



ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

PLAN D'ÉTUDES INFORMATIQUE

2005 - 2006

arrêté par la direction de l'EPFL le 6 juin 2005

Directeur de la section	Prof. A. Schiper
Directeur adjoint de la section	Prof. W. Gerstner
Conseillers d'études :	
1ère année	Prof. P. Fua
2ème année	Prof. C. Petitpierre
3ème année	Prof. P. Ienne
4ème année	Prof. Th. Henzinger
Diplômants	Prof. R. Hersch
Responsable passerelle HES	Prof. A. Schiper
Coordinateur SHS	Prof. A. Wegmann
Délégué à la mobilité	Dr. M. Lundell
Secrétariat Bachelor	Mme Cecilia Bigler
Secrétariat Master	Mme Kathleen Collins
Administratrice de la section	Mme Sylviane Dal Mas

Au cycles Bachelor et master, selon les besoins pédagogiques, les heures d'exercices mentionnées dans le plan d'études pourront être intégrées dans les heures de cours ; les scolarités indiquées représentent les nombres moyens d'heures de cours et d'exercices hebdomadaires sur le semestre.

INFORMATIQUE

Cycle propédeutique

Matières	Les enseignants sont indiqués sous réserve de modification	Sections	Semestres						Coeff.	Période des épreuves	Type exam.
			1			2					
			c	e	p	c	e	p			
Bloc "Branches d'examen" :											
Analyse I,II (en français) ou	Bachmann	MA	4	4		4	2		15	E ou A	écrit
Analyse I,II (en allemand)	Semmler	MA	4	4		4	2		5	E ou A	écrit
Algèbre linéaire	Cibils	MA	4	2					3	E ou A	écrit
Informatique théorique I,II	Bartholdi	MA	2	2		2	2		4	E ou A	écrit
Physique générale I	Meylan	PH				4	2		3	E ou A	écrit
Bloc "Branches de semestre" :											
Electronique I,II	Zysmann	EL	2	1		2	1	2	4	sem H + E	
Introduction aux systèmes informatiques	Sanchez	IN	2	1					2	sem H	
Programmation I+II	Sam + vacat	IN/--	2	2	2	2	2	2	4	sem H + E	
Systèmes logiques	Sanchez	IN				2	2		2	sem E	
SHS : Cours d'initiation 1	Divers enseignants	SHS	1						0.25	sem H	
SHS : Cours d'initiation 2	Divers enseignants	SHS	1						0.25	sem H	
SHS : Cours d'initiation 3	Divers enseignants	SHS				1			0.25	sem E	
SHS : Cours d'initiation 4	Divers enseignants	SHS				1			0.25	sem E	
Totaux :			18	12	2	18	7	6			
Totaux : Par semaine			32			31					

Légende :

c : cours e : exercices p : branches pratiques
 colonnes c/e/p : nb d'heures par semaine
 1 semestre comprend 14 semaines.
 type examination : voir règlement d'application

en italique : cours à option
 () : facultatif
 / : enseignement partagé
 + : enseignement séparé à l'horaire

INFORMATIQUE

Cycle Bachelor

Matières	Les enseignants sont indiqués sous réserve de modification	Sections	Semestres												Crédits		Période des épreuves	Type exam.	
			3			4			5			6			2ème	3ème			
			c	e	p	c	e	p	c	e	p	c	e	p					
Bloc A																28			
Analyse III	Liebendörfer	MA	3	2												5		H	écrit
Analyse numérique	Burman	MA				2	1									3		E	écrit
Bases de données	Spaccapietra	IN				2	2	2								5		E	écrit
Physique générale II	Félix	PH	4	2												6		H	écrit
Probabilité et statistique I,II	Mountford	MA	2	1		2	1									6		E ou A	écrit
Programmation Internet	Petitpierre	IN				2		2								3		sem E	
Bloc B																28			
Algorithmique	Shokrollahi	MA				4	2									6		E	écrit
Architecture des ordinateurs I,II	Jenne	IN	2			2	2	2								8		sem H+E	
Programmation III + IV	Chappelier + Odersky	IN	2			2	2	2								8		sem H+E	
Theoretical computer science III	Henzinger	IN	4	2												6		H	écrit
Bloc C																29			
Concurrence	Sandoz	SC							2	1						3		H	écrit
Informatique du temps réel	Decotignie	SC							2	1						3		H	écrit
Introduction au marketing et à la finance	Wegmann/Schwab	SC										2				2		sem E	
Mathématiques discrètes	Hêche	MA							2	1						3		H	écrit
Recherche opérationnelle	Hêche	MA										2	1			3		E	écrit
Réseaux informatiques	Grossglauser	SC										2	2			4		E	écrit
Systèmes d'exploitation	Sandoz	SC										2	1			4		E	écrit
Systèmes répartis	Schipper	SC										2	1			4		E	écrit
Théorie de l'information	Chappelier	IN										2	1			3		E	écrit
Bloc D																27			
Compiler construction	Odersky	IN							2	2	2					6		sem H	
Computer graphics	Thalmann	IN							2		1					3		H	écrit
Intelligence artificielle	Faltings	IN										2		2		4		sem E	
Projet de génie logiciel I,II	Hulaas/Petitpierre	IN										5		5		10		sem H+E	
Software engineering	Baar	IN							4							4		sem H	
Enseignement sciences humaines et sociales (SHS)																8			
SHS : Atelier I,II	Divers enseignants	SHS			2			2								4		sem H+E	
SHS : Cours de spécialisation I,II	Divers enseignants	SHS							2			2				4		sem H+E	
Totaux : Tronc commun			17	7	6	16	6	10	16	5	8	16	6	7		60	60		
Totaux : Par semaine			30			32			29			29							

Légende :

c : cours e : exercices p : branches pratiques
 colonnes c/e/p : nb d'heures par semaine
 1 semestre comprend 14 semaines.
 type examination : voir règlement d'application

en italique : cours à option
 () : facultatif
 / : enseignement partagé
 + : enseignement séparé à l'horaire

INFORMATIQUE - Options et spécialisations

Matières	Les enseignants sont indiqués sous réserve de modification	Sections	Spécialisation	Semestres						Crédits	Période des épreuves	Type examen	cours biennaux / donnés en
				M1			M2						
	Enseignants			c	e	p	c	e	p		indiqués sous réserve de modification		
Spécialisations :													
1 BIOCUMPETING			1										
2 FOUNDATIONS OF SOFTWARE			2										
3 IMAGE SCIENCE			3										
4 INDUSTRIAL INFORMATICS			4										
5 INTERNET COMPUTING			5										
6 COMPUTER ENGINEERING			6										

Options

Advanced compiler construction	Schinz	IN	2				2	2		4	E	oral	
Advanced computer architecture	Ienne	IN	6				2	2		4	E	oral	
Advanced computer graphics	Thalmann	IN	3				2	1		4	E	écrit	
Advanced databases	Spaccapietra	IN	5	2	2	2				6	sem H		
Advanced digital design	Sanchez	IN	6				4	2		6	sem E		
Algorithms (pas donné en 2005-2006)	Shokrollahi	MA	2 5	4	2	1				7	H	écrit	
Color reproduction	Hersch	IN	3				2	2		4	E	oral	
Combinatoire	Prodon	MA					2	2		4	E	oral	2006-2007
Complex circuits	Beuchat/Piguet	IN	6	2		2				4	H	oral	
Computational genomics	Galisson	IN	1	4	2					6	H	écrit	
Computational processing of textual data	Rajman/Chappelier	IN	5				4	2		6	E	écrit	
Computer vision	Fua	IN	3				2	1		4	E	écrit	
Computer-aided verification	Henzinger	IN	2 6				4		2	6	sem E		
Computer-Supported Cooperative Work (CSCW)	Dillenbourg	IN		2	2					6	H	oral	
Cryptography and security	Vaudenay/Oechslin	SC	5	4		2				7	H	écrit	
Design technologies for intergrated systems	De Micheli	IN	6	3		2				6	sem H		
Distributed algorithms	Schipper	SC	2 5	2	1					4	H	écrit	
Distributed information systems	Aberer	SC	1 5	2	1					4	H	écrit	
Dynamical system theory for engineers	Hasler/DeFeo	SC	1	4	2					7	H	écrit	
Embedded systems	Beuchat	IN	4 6				2		2	4	E	oral	
Enterprise architecture and systems engineering (dès 2006-2007)	Wegmann	SC	4 5	4	2					6	H	écrit	
Foundations of image science	Fua	IN	3	2	1					4	H	écrit	
Genetic and developmental computing architectures	Tempesti / Ijspeert	IN		4	2					6	H	à fixer	
Graphes et réseaux I, II	de Werra	MA		2	2		2	2		8	E	oral	2005-2006
Human computer interaction	Pu	IN	5				2	1		4	sem E		
Industrial automation	Kirrmann	SC	4				2		1	3	E	oral	
Intelligent agents	Faltings	IN	4 5	3	3					6	H	écrit	
Middleware (pas donné en 2005-2006)	vacat	SC	5				4	2		7	E	écrit	
Mobile networks	Hubaux	SC	5	2	1					4	H	écrit	
Modelling the immune system	Kraehenbuhl/Le Boudec/Martinoli	SC	1	1	1					3		oral	
Models of biological sensory-motor systems	Ijspeert	IN	1	2		2				4	H	à fixer	
Multimedia documents	Vanoirbeek	IN	5				4	2		6	E	écrit	
Optimisation I, II	de Werra	MA		2	2		2	2		8	E	oral	2006-2007
Parallélisation de programmes sur grappes de PC	Hersch	IN		1		2				3	sem H		
Pattern classification and machine learning	Gerstner/Hasler	IN/SC	1 3				4	2		6	E	écrit	
Performance evaluation	Le Boudec	SC	2 4 5				4	2		7	E	oral	
Périphériques	Gerlach/Hersch	IN		2		1				3	H	oral	
Real-time embedded systems	Beuchat	IN	6	2	2					4	sem H		
Real-time programming	Decotignie	SC	6	3		1				4	H	écrit	
Real-time systems	Decotignie	SC	4				2			3	E	écrit	
Selected topics in distributed computing (pas donné en 2005-2006)	Guerraoui	SC	2 5	2	1					4	H	écrit	
Student seminar : AI methods for biology	Faltings	IN	1	1	1					2	sem H		2006-2007
Student seminar : Information systems in biology	Aberer/Palaghi	SC	1				1	1		2	sem E		2006-2007
Swarm intelligence	Martinoli	SC	1 6	2		3				6	H	oral	
Traitement automatique de la parole	Boulevard	EL		2	1					3	H	écrit	
Type systems (Pas donné en 2005/2006)	vacat	IN	2	2		2				4	H	oral	
Unsupervised and reinforcement learning in neural networks	Gerstner	IN	1	2	2					4	H		2006-2007
Virtual reality	Thalmann	IN	3				2	1		4	E	écrit	

Légende :

c : cours e : exercices p : branches pratiques
 colonnes c/e/p : nb d'heures par semaine
 1 semestre comprend 14 semaines.
 type examination : voir règlement d'application

en italique : cours à option
 () : facultatif
 / : enseignement partagé
 + : enseignement séparé à l'horaire

Règles d'admission en 2^{ème} année Informatique pour les étudiants ayant fait leur 1^{ère} année dans une autre section de l'EPFL

1. Les étudiants de systèmes de communication ayant réussi leur première année peuvent changer de section sans rattrapage.

2. Les étudiants EPFL des autres sections (à l'exception des architectes) qui ont réussi leur première année peuvent entrer en 2^{ème} année à condition de rattraper
 - Electronique I, II
 - Informatique théorique I, II
 - Introduction aux systèmes informatiques
 - Programmation I+II
 - Systèmes logiques

Cependant, si la section d'origine a en première année un enseignement en mathématique comparable à celui dispensé en Informatique
et
si la section d'origine dispense en première année au moins un cours en informatique
et
si l'étudiant a au moins 5 de moyenne en mathématique et 5 de moyenne en informatique,
alors l'étudiant peut entrer en deuxième année sans rattrapage (*).

Dans ce dernier cas, l'étudiant doit être conscient qu'il aura à faire un effort personnel de rattrapage des connaissances qui lui manquent.

(*) Les sections qui entrent dans cette catégorie sont :

- Génie électrique et électronique
- Génie mécanique
- Microtechnique
- Physique
- Mathématiques
- Sciences et technologies du vivant
- Science et génie des matériaux

Les sections qui n'entrent pas dans cette catégorie sont :

- Architecture
- Chimie
- Génie civil
- Sciences et ingénierie de l'environnement

**RÈGLEMENT D'APPLICATION DU CONTRÔLE DES
ÉTUDES DE LA SECTION
D'INFORMATIQUE
(sessions de printemps, d'été et d'automne 2006)
du 6 juin 2005**

La direction de l'Ecole polytechnique fédérale de Lausanne

vu l'ordonnance sur la formation menant au bachelor et au master de l'EPFL, du 14 juin 2004,

vu l'ordonnance sur le contrôle des études menant au bachelor et au master à l'EPFL, du 14 juin 2004,

arrête

Article premier - Champ d'application

Le présent règlement est applicable aux examens de la section d'Informatique de l'EPFL dans le cadre des études de bachelor et de master.

Art. 2 – Etapes de formation

1 Le bachelor est composé de deux étapes successives de formation :

- le cycle propédeutique d'une année dont la réussite se traduit par 60 crédits ECTS acquis en une fois, condition pour entrer au cycle bachelor.
- le cycle bachelor s'étendant sur deux ans dont la réussite implique l'acquisition de 120 crédits, condition pour entrer au master.

2 Le master est composé de deux étapes successives de formation :

- le cycle master d'une durée d'un an dont la réussite implique l'acquisition de 60 crédits.
- le projet de master d'une durée de 4 mois dont la réussite implique l'acquisition de 30 crédits.

Art. 3 – Sessions d'examen

1 Les branches d'examen sont examinées par écrit ou par oral pendant les sessions de printemps, été ou automne. Elles sont mentionnées dans le plan d'études avec la mention H, E ou A.

2 Les branches de semestre sont examinées pendant le semestre d'hiver ou le semestre d'été. Elles sont mentionnées dans le plan d'études avec la mention sem H ou sem E.

3 Une branche annuelle (c'est-à-dire dont l'intitulé tient sur une seule ligne dans le plan d'études) est examinée globalement pendant la session d'été (E). Le plan d'études précise si la session d'automne (A) peut remplacer la session d'été.

Chapitre 1 : Cycle propédeutique

Art. 4 - Examen propédeutique

1. L'examen propédeutique est composé du bloc des branches d'examen et du bloc des branches de semestre.
2. Les modalités et les conditions de réussite sont fixées par le chapitre 2 de l'ordonnance sur le contrôle des études menant au bachelor et au master à l'EPFL.

Chapitre 2 : Cycle bachelor

Art. 5 - Organisation

Les enseignements du cycle bachelor sont répartis entre la 2^e année et la 3^e année de la façon suivante:

- blocs A et B + SHS pour la 2^{ème} année ;
- blocs C et D + SHS pour la 3^{ème} année.

Art. 6 - Examen de 2^{ème} année

1 Le bloc « A » est réussi lorsque les **28 crédits** du plan d'études sont obtenus.

2 Le bloc « B » est réussi lorsque les **28 crédits** du plan d'études sont obtenus.

Art. 7 - Examen de 3^{ème} année

1 Le bloc « C » est réussi lorsque les **29 crédits** du plan d'études sont obtenus.

4 Le bloc « D » est réussi lorsque les **27 crédits** du plan d'études sont obtenus.

Art. 8 - Examen de 2^{ème} et 3^{ème} année

Le bloc « SHS transversal » est réussi lorsque les **8 crédits** du plan d'études sont obtenus.

Chapitre 5 : Cycle master

Art. 9 - Organisation

1 Les enseignements du master sont répartis en 1 bloc "Projets + SHS" et en 1 groupe "cours à option" dont les crédits doivent être obtenus de façon indépendante.

Art. 10 - Cours à option

1 Des cours, comptant pour un maximum de 15 crédits au total, peuvent être choisis en dehors de la liste des cours à option définis dans la partie « master » du plan d'étude.

2 Si ces cours ne font pas partie d'une spécialisation, ils doivent être acceptés préalablement par le directeur de la section qui fixe le nombre de crédits à leur attribuer.

Art. 11 - Examen du cycle master

1 Le bloc "Projet + SHS" est réussi lorsque les **18 crédits** du plan d'études sont obtenus.

2 Le groupe de cours à option est réussi lorsque les **42 crédits** sont obtenus de façon indépendante, par réussite individuelle de chaque branche.

Art. 12 – Spécialisations et mineurs

1 Le Master est décerné avec un mineur de spécialisation si l'étudiant a acquis 30 crédits de cours pris dans la liste des cours de la spécialisation et s'il a réussi le projet de master.

2 Le Master est décerné avec un mineur externe si l'étudiant a acquis 30 crédits d'un programme de mineur et s'il a réussi le projet de master.

3 Il est possible d'effectuer un mineur de spécialisation et un mineur externe mineur en effectuant 30 crédits supplémentaires.

4 Il n'est pas possible de faire deux spécialisations.

Chapitre 5 : Dispositions finales

Art. 13 - Abrogation du droit en vigueur

Le règlement d'application du contrôle des études de la section d'Informatique de l'EPFL du 24 mai 2004 est abrogé.

Art. 14 - Entrée en vigueur

Le présent règlement est applicable aux examens correspondant au plan d'études 2005/2006.

Au nom de la direction de l'EPFL

Le président, P. Aebischer
Le vice-président pour les affaires académiques,
G. Margaritondo

Lausanne, le 6 juin 2005

SPÉCIALISATIONS (SIN)

1. <u>BIOCOMPUTING</u> : (total crédits : 73)	<u>crédits</u>	<u>page</u>
Analyse de données génétiques (donné en 2005-2006)	(4)	133
Computational genomics	(6)	94
Distributed information systems	(4)	102
Dynamical system theory for engineers	(7)	103
Infochimie	(4)	144
Mathematical modelling of DNA I	(4)	145
Mathematical modelling of DNA II	(4)	146
Modelling the immune system	(3)	113
Models of biological sensory-motor systems	(4)	114
Pattern classification and machine learning	(6)	117
Réseaux de neurones et modélisation biologique	(3)	152
Swarm intelligence	(6)	127
Traitement d'images I	(3)	155
Traitement d'images II	(3)	156
Statistics for genomics data analysis	(4)	<i>(pas donné en 05-06)</i>
Student seminar : AI methods for biology	(2)	<i>(pas donné en 05-06)</i>
Student seminar : Information systems in biology	(2)	<i>(pas donné en 05-06)</i>
Unsupervised and reinforcement learning in neural networking	(4)	<i>(pas donné en 05-06)</i>
2. <u>FOUNDATIONS OF SOFTWARE</u> (total crédits : 35) (seul. 21 en 05-06)	<u>crédits</u>	<u>page</u>
Advanced compiler construction	(4)	87
Computer-aided verification	(6)	96
Distributed algorithms	(4)	101
Performance evaluation	(7)	118
Algorithms	(6)	<i>(pas donné en 05-06)</i>
Selected topics in distributed computing	(4)	<i>(pas donné en 05-06)</i>
Type systems	(4)	<i>(pas donné en 05-06)</i>
3. <u>IMAGE SCIENCE</u> (total crédits : 30)	<u>crédits</u>	<u>page</u>
Advanced Computer Graphics	(4)	89
Color imaging	(4)	136
Color reproduction	(4)	92
Computer vision	(4)	98
Foundations of image science	(4)	105
Pattern classification and machine learning	(6)	117
Virtual reality	(4)	129

4. <u>INDUSTRIAL INFORMATICS</u> (total crédits : 64)	<u>crédits</u>	<u>page</u>
Automatique I	(3)	134
Automatique II	(3)	135
Embedded systems	(4)	104
Enterprise architecture and systems engineering (<i>dès 2006/2007</i>)	(6)	
Gestion de production I,II	(4)	138/139
Identification et commande I	(2)	142
Identification et commande II	(2)	143
Industrial automation	(3)	110
Intelligent agents	(6)	111
Mécatronique	(2)	147
Modèles stochastiques pour les communications	(6)	148
Optimisation I	(3)	149
Optimisation II	(3)	150
Performance evaluation	(7)	118
Real-time systems	(3)	126
Recherche opérationnelle	(3)	151
Systèmes multivariables I	(2)	153
Systèmes multivariables II	(2)	154
5. <u>INTERNET COMPUTING</u> (total crédits : 74)	<u>crédits</u>	<u>page</u>
Advanced databases	(6)	90
Computational processing of textual data	(6)	95
Cryptography and Security	(7)	99
Distributed algorithms	(4)	101
Distributed information systems	(4)	102
E-Business	(6)	137
Enterprise architecture and systems engineering (<i>dès 2006/2007</i>)	(6)	
Human-computer interaction	(4)	109
Intelligent agents	(6)	111
Mobile networks	(4)	112
Multimedia documents	(6)	115
Performance evaluation	(7)	118
Algorithms	(7)	<i>(pas donné en 05-06)</i>
Middleware	(7)	<i>(pas donné en 05-06)</i>
6. <u>COMPUTER ENGINEERING</u> (total crédits : 48)	<u>crédits</u>	<u>page</u>
Advanced computer architecture	(4)	88
Advanced digital design	(6)	91
Complex circuits	(4)	93
Computer-aided verification	(6)	96
Design technologies for integrated systems	(6)	100
Embedded systems	(4)	104
Hardware systems modeling I	(2)	140
Hardware systems modeling II	(2)	141
Real-time embedded systems	(4)	124
Swarm intelligence	(6)	127
VLSI design I	(2)	157
VLSI design II	(2)	158



ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

SECTION D'INFORMATIQUE

Cycle

Propédeutique

(1ère année)

2005 / 2006

Titre	Algèbre linéaire
Title	Linear Algebra

Enseignant(s) / Instructor(s)	Cibils Michel: MA	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Informatique (2005-2006, Bachelor semestre 1)	Cours: 4 Exercice: 2		
Génie électrique et électronique (2005-2006, Bachelor semestre 1)	Cours: 4 Exercice: 2		

Objectifs:

Apprendre les techniques du calcul matriciel, être apte à exécuter les manipulations mathématiques s'y rapportant et être capable d'appliquer ces techniques dans les problèmes issus de son domaine de spécialisation.
L'étudiant devra maîtriser les outils nécessaires à la résolution des problèmes liés à la linéarité, à l'orthogonalité et à la diagonalisation des matrices.

Contenu:

- Système d'équations linéaires
 - Calcul matriciel
 - Déterminants
 - Espaces vectoriels
 - Valeurs et vecteurs propres
 - Orthogonalité et moindres carrés
 - Matrices symétriques et formes quadratiques.
- Le cours est illustré d'exemples pratiques du domaine des sciences de l'ingénieur.
Les exercices sont réalisés à l'aide du logiciel Matlab.

Préparation pour:

Analyse II et III

Forme d'enseignement:

Exposé oral, exercices en salle d'ordinateurs

Forme du contrôle:

Contrôle continu : exercices chaque semaine et travaux écrits

Bibliographie:

Obligatoire
Linear Algebra and its Applications,
D.C. Lay, 3rd edition (or updated 2nd edition) Addison-Wesley.

Objectives:

Learn the techniques of matrix algebra, be able to execute the corresponding mathematical manipulations and to apply these techniques in problems connected to one's specialization area.
The student will have to master the tools necessary to the resolution of problems connected to linearity, orthogonality and matrix diagonalization.

Content:

- Systems of linear equations
 - Matrix Algebra
 - Determinants
 - Vector Spaces
 - Eigenvalues and eigenvectors
 - Orthogonality and least-squares
 - Symmetric matrices and quadratic forms.
- The course is illustrated by examples coming from the area of technical sciences.
Exercises are done with the help of the software Matlab.

Matière examinée / subjects examined		Algèbre linéaire			
Session	ETE AUT	Coefficient / Crédits ECTS	3	Forme de l'examen / Form of examination	Écrit

Titre	Analyse I
Title	Analysis I

Enseignant(s) / Instructor(s)	Bachmann Otto: MA	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Informatique (2005-2006, Bachelor semestre 1)	Cours: 4 Exercice: 4		
Systèmes de communication (2005-2006, Bachelor semestre 1)	Cours: 4 Exercice: 4		

Objectifs:

Etude des méthodes principales du calcul différentiel et intégral en vue de leur utilisation par les ingénieurs.

Contenu:

Calcul différentiel et intégral des fonctions d'une variable.
 - Notions fondamentales (nombres réels et complexes, suites, séries, limite)
 - Fonctions d'une variable (limite, continuité et dérivée)
 - Développements limités
 - Comportement local d'une fonction, extréma
 - Fonctions particulières (logarithme, exponentielle, trigonométrique et hyperbolique)
 - Intégrales.

Forme d'enseignement:

Ex cathedra et exercices en salle

Forme du contrôle:

Bibliographie:

Donnée au cours

Objectives:

Study of the principal methods of calculus in view of its applications by engineers.

Content:

Diferential and integral calculus of one variable
 - Fundamental nations (real and complex numbers, sequences, series, limits)
 - Functions of one variable (limite, continuity and derivability)
 - Local behavior of a function, maxima and minima
 - Special functions (logarithm, exponential, trigonometric, hyperbolic)
 - Integrals

Form of teaching:

Ex cathedra and exercises in class

Matière examinée / subjects examined	Analyse I,II		
Session	ETE AUT	Coefficient / Crédits ECTS	5
Forme de l'examen / Form of examination			Ecrit

Titre	Analyse I (allemand)
Title	Analysis I

Enseignant(s) / Instructor(s)		Semmler Klaus-Dieter: MA		Langue / Language	DE
Programme(s) Période(s)		Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours	
Informatique (2005-2006, Bachelor semestre 1)		Cours: 4 Exercice: 4			
Génie civil (2005-2006, Bachelor semestre 1)		Cours: 4 Exercice: 2			
Sciences et ingénierie de l'environnement (2005-2006, Bachelor semestre 1)		Cours: 4 Exercice: 2			
Génie électrique et électronique (2005-2006, Bachelor semestre 1)		Cours: 4 Exercice: 4			
Génie mécanique (2005-2006, Bachelor semestre 1)		Cours: 4 Exercice: 4			
Mathématiques (2005-2006, Bachelor semestre 1)		Cours: 4 Exercice: 4			
Science et génie des matériaux (2005-2006, Bachelor semestre 1)		Cours: 4 Exercice: 4			
Microtechnique (2005-2006, Bachelor semestre 1)		Cours: 4 Exercice: 4			
Physique (2005-2006, Bachelor semestre 1)		Cours: 4 Exercice: 4			
Sciences et technologies du vivant (2005-2006, Bachelor semestre 1)		Cours: 4 Exercice: 4			
Systèmes de communication (2005-2006, Bachelor semestre 1)		Cours: 4 Exercice: 4			
Enseignement maths - chimie, biologie, géosciences (2005-2006, Master semestre 1)		Cours: 4 Exercice: 4			

Objectifs:

Cours de base en allemand, orienté vers les applications et les besoins de l'ingénieur.

Contenu:

- Nombres réels
- Suites et séries
- Fonctions, limites et continuité
- Nombres complexes
- Calculs différentiels des fonctions de IR en IR
- Intégration, primitives
- Intégrales généralisées
- Equations différentielles de premier et deuxième ordre.

Préparation pour:

Analyse II

Forme d'enseignement:

Cours, exercices en groupes. Le vocabulaire mathématique sera travaillé de façon bilingue (a/f)

Forme du contrôle:

Exercices à rendre.

Bibliographie:

Sera communiquée au cours (Policopié).

Matière examinée / subjects examined		Analyse I,II (allemand)			
Session	ETE AUT	Coefficient / Crédits ECTS	5	Forme de l'examen / Form of examination	Ecrit

Titre	Analyse II
Title	Analysis II

Enseignant(s) / Instructor(s)	Bachmann Otto: MA	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Informatique (2005-2006, Bachelor semestre 2)	Cours: 4 Exercice: 2		
Systèmes de communication (2005-2006, Bachelor semestre 2)	Cours: 4 Exercice: 2		

Objectifs:

Etude des méthodes principales du calcul différentiel et intégral en vue de leur utilisation par les ingénieurs.

Contenu:

Eléments d'équations différentielles ordinaires.
 - Équations différentielles du premier ordre
 - Équations différentielles du deuxième ordre à coefficients constants
 Calcul différentiel et intégral des fonctions de plusieurs variables.
 - Fonctions de plusieurs variables
 - Dérivées partielles
 - Différentielle
 - Extrema
 - Intégrales multiples
 - Intégrales curvilignes.

Prérequis:

Analyse I, Algèbre linéaire I

Forme d'enseignement:

Ex cathedra et exercices en salle

Bibliographie:

Donnée au cours.

Objectives:

Study of the principal methods of calculus in view of its applications by engineers.

Content:

Introduction to the theory of ordinary differential equations
 - First order differential equations
 - Second order differential equations with constant coefficients
 Differential and integral calculus of several variables
 - Multivariable functions
 - Partial derivatives
 - Differentials
 - Maxima and minima
 - Multiple integrals
 - Line integrals

Required prior knowledge:

Analyse I, Algèbre linéaire I

Form of teaching:

Ex cathedra and exercises in class

Matière examinée / subjects examined	Analyse I,II		
Session	ETE AUT	Coefficient / Crédits ECTS	5
Forme de l'examen / Form of examination			Ecrit

Titre	Analyse II (allemand)
Title	Analysis II

Enseignant(s) / Instructor(s)		Semmler Klaus-Dieter: MA		Langue / Language	DE
Programme(s) Période(s)		Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours	
Informatique (2005-2006, Bachelor semestre 2)		Cours: 4 Exercice: 2			
Génie civil (2005-2006, Bachelor semestre 2)		Cours: 4 Exercice: 2			
Sciences et ingénierie de l'environnement (2005-2006, Bachelor semestre 2)		Cours: 4 Exercice: 2			
Génie électrique et électronique (2005-2006, Bachelor semestre 2)		Cours: 4 Exercice: 2			
Génie mécanique (2005-2006, Bachelor semestre 2)		Cours: 4 Exercice: 2			
Mathématiques (2005-2006, Bachelor semestre 2)		Cours: 4 Exercice: 4			
Science et génie des matériaux (2005-2006, Bachelor semestre 2)		Cours: 4 Exercice: 2			
Microtechnique (2005-2006, Bachelor semestre 2)		Cours: 4 Exercice: 2			
Physique (2005-2006, Bachelor semestre 2)		Cours: 4 Exercice: 4			
Sciences et technologies du vivant (2005-2006, Bachelor semestre 2)		Cours: 4 Exercice: 2			
Systèmes de communication (2005-2006, Bachelor semestre 2)		Cours: 4 Exercice: 2			
Enseignement maths - chimie, biologie, géosciences (2005-2006, Master semestre 2)		Cours: 4 Exercice: 4			

Objectifs:

Cours de base en allemand, orienté vers les applications et les besoins de l'ingénieur.

Contenu:

- Calculs différentiels des fonctions de IR^n en IR^m
- Limites, continuité, extréma
- Gradient, dérivée directionnelle, points critiques
- Formes différentielles, facteurs intégrantes, intégrales curvilignes
- Intégration sur des domaines en IR^n
- Formule de Green-Stokes.

Prérequis:

Analyse I

Forme d'enseignement:

Cours, exercices en groupes. Le vocabulaire mathématique sera travaillé de façon bilingue (a/f)

Bibliographie:

Sera communiquée au cours (Polycopié).

Matière examinée / subjects examined		Analyse I,II (allemand)			
Session	ETE AUT	Coefficient / Crédits ECTS	5	Forme de l'examen / Form of examination	Ecrit

Titre	Electronique I
Title	Electronics I

Enseignant(s) / Instructor(s)	Zysman Eytan: EL	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Informatique (2005-2006, Bachelor semestre 1)	Cours: 2 Exercice: 1		

Objectifs:

A la fin du cours, l'étudiant sera capable de comprendre et de concevoir correctement les circuits électroniques de base. Cet objectif s'appuie sur une connaissance fondamentale des composants électroniques modernes et la maîtrise de leur mise en oeuvre dans les circuits. L'étudiant aura une approche théorique et également "physique" des phénomènes et des techniques de circuits et saura interpréter des résultats de calcul ou de mesures. Il aura le sens des approximations et leurs limites de validité.

Contenu:

Cours

1. Circuits passifs linéaires
2. Circuits passifs non-linéaires
3. Amplificateur opérationnel en contre-réaction
4. Amplificateur opérationnel en réaction positive
5. Imperfections des amplificateurs opérationnels
6. Applications de l'amplificateur opérationnel
7. Oscillateurs
8. Bascules

Exercices et travaux pratiques

Avec les exercices et travaux pratiques, l'étudiant confrontera systématiquement la théorie aux résultats expérimentaux. Il mettra en oeuvre différents types de circuits intégrés et de composants discrets dans diverses expériences.

Prérequis:

Electrotechnique I et II

Préparation pour:

Electronique II

Forme d'enseignement:

Cours ex cathedra et exercices dirigés en salle. Travaux pratiques en laboratoire

Bibliographie:

Notes de cours polycopiées. Notice de laboratoire

Objectives:

At the end of the course, the student will be able to understand and design basic electronics. This objective takes advantage of an in-depth knowledge of modern electronic components and their applications. The student will acquire both theoretical and physical approaches of phenomena as well as practical aspects of design limitations and measurements of circuits.

Content:

Courses

1. Linear passive circuits
2. Non-linear passive circuits
3. Op.-Amp. with negative feed-back
4. Op.-Amp. with positive feed-back
5. Non-ideal effects in Op.-Amp.
6. Op.-Amp. applications
7. Oscillators
8. Triggers and timers

Exercises and laboratories

Exercises and laboratory experiments will allow the student to compare theory and practice. Different types of integrated circuits as well as discrete components will be used in various experiments.

Required prior knowledge:

Electrotechnics I and II

Prerequisite for:

Electronics II

Form of teaching:

Ex cathedra and exercises in class. Labs

Form of examination:

Written

Matière examinée / subjects examined		Electronique I,II			
Session	ETE	Coefficient / Crédits ECTS	4	Forme de l'examen / Form of examination	Ctrl continu

Titre	Electronique II
Title	Electronics II

Enseignant(s) / Instructor(s)	Zysman Eytan: EL	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Informatique (2005-2006, Bachelor semestre 2)	Cours: 2 Exercice: 1 TP: 2 TP: 2		

Objectifs:

A la fin du cours, l'étudiant sera capable de comprendre et de concevoir correctement les circuits électroniques de base. Cet objectif s'appuie sur une connaissance fondamentale des composants électroniques modernes et la maîtrise de leur mise en oeuvre dans les circuits. L'étudiant aura une approche théorique et également "physique" des phénomènes et des techniques de circuits et saura interpréter des résultats de calcul ou de mesures. Il aura le sens des approximations et leurs limites de validité.

Contenu:

Cours

- 9. Semiconducteurs et jonction pn
- 10. Diode
- 11. Transistor bipolaire
- 12. Transistor MOS
- 13. Configurations petits signaux du transistor
- 14. Polarisation et sources de courant
- 15. Amplificateurs élémentaires à transistors
- 16. Réponse en fréquence des amplificateurs

Exercices et travaux pratiques

Avec les exercices et travaux pratiques, l'étudiant confrontera systématiquement la théorie aux résultats expérimentaux. Il mettra en oeuvre différents types de circuits intégrés et de composants discrets dans diverses expériences.

Prérequis:

Electronique I

Préparation pour:

Circuits et Systèmes Electroniques

Forme d'enseignement:

Cours ex cathedra et exercices dirigés en salle. Travaux pratiques en laboratoire

Forme du contrôle:

Ecrit

Bibliographie:

Notes de cours polycopiées. Notice de laboratoire

Objectives:

At the end of the course, the student will be able to understand and design basic electronics. This objective takes advantage of an in-depth knowledge of modern electronic components and their applications. The student will acquire both theoretical and physical approaches of phenomena as well as practical aspects of design limitations and measurements of circuits.

Content:

Courses

- 9. Semiconductors and pn junction
- 10. Diode
- 11. Bipolar transistor
- 12. MOS transistor
- 13. Small signal configurations
- 14. Bias and current sources
- 15. Basic amplifiers
- 16. Frequency response of amplifiers

Exercises and laboratories

Exercises and laboratory experiments will allow the student to compare theory and practice. Different types of integrated circuits as well as discrete components will be used in various experiments.

Required prior knowledge:

Electronics I

Prerequisite for:

Electronic circuits and systems

Form of teaching:

Ex cathedra and exercises in class. Labs

Form of examination:

Written

Matière examinée / subjects examined		Electronique I,II			
Session	ETE	Coefficient / Crédits ECTS	4	Forme de l'examen / Form of examination	Ctrl continu

Titre	Informatique théorique I
Title	Theoretical computer science I

Enseignant(s) / Instructor(s)	Bartholdi Laurent: MA	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Informatique (2005-2006, Bachelor semestre 1)	Cours: 2 Exercice: 2		

Objectifs:

Introduire les outils et le langage mathématique nécessaire à l'étude théorique de l'informatique.

Apprendre les techniques de manipulation de concepts mathématiques fondamentaux.

Contenu:

Le cours introduira la logique élémentaire et les notations mathématiques courantes jusqu'à Noël. Il continuera avec les chapitres 1-2 du livre, vus à la fois en classe et aux exercices.

Les exercices consisteront entre autres en une lecture personnelle de certaines sections du livre.

1. Logique élémentaire
2. Ensembles, relations, fonctions
3. Logique propositionnelle du 1er ordre
4. Récurrences
5. Sommmations.

Préparation pour:

Toute la partie théorique du plan d'études

Forme d'enseignement:

Cours ex cathedra avec exercices et lecture personnelle du livre

Forme du contrôle:

Branche d'examen (écrit)

Bibliographie:

« Concrete Mathematics », requis, ISBN 0-201-55802-5 (anglais) ou 2-8418-0981-1 (français).

Objectives:

Introduce the tools and the mathematical language necessary to the theoretical study of computer science.

Learn the manipulation techniques of fundamental mathematical concepts.

Content:

The course will introduce elementary logic and standard mathematical notations till Christmas. It will continue with chapters 1-2 of the book, studied both in class and in the exercise sessions.

These exercise sessions will consist among others in a personal reading of selected sections of the book.

1. Elementary Logic
2. Sets, relations, functions
3. First order propositional logic
4. Recurrences
5. Summation.

Matière examinée / subjects examined		Informatique théorique I,II	
Session	ETE AUT	Coefficient / Crédits ECTS	4
Forme de l'examen / Form of examination			Ecrit

Titre	Informatique théorique II
Title	Theoretical computer science II

Enseignant(s) / Instructor(s)	Bartholdi Laurent: MA	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Informatique (2005-2006, Bachelor semestre 2)	Cours: 2 Exercice: 2		

Objectifs:

Introduire les outils et le langage mathématique nécessaire à l'étude théorique de l'informatique.

Apprendre les techniques de manipulation de concepts mathématiques fondamentaux.

Contenu:

Le cours suivra les chapitres 3-8 du livre, à la fois en classe et en petits groupes de travail.

Les exercices consisteront entre autres en une lecture personnelle de certaines sections du livre.

1. Fonctions entières
2. Théorie des nombres, cryptographie
3. Identités binomiales
4. Nombres remarquables
5. Fonctions génératrices
6. Probabilités discrètes.

Prérequis:

Informatique Théorique I

Préparation pour:

Toute la partie théorique du plan d'études

Forme d'enseignement:

Cours ex cathedra avec exercices et lecture personnelle du livre

Forme du contrôle:

Branche d'examen (écrit)

Bibliographie:

« Concrete Mathematics », requis, ISBN 0-201-55802-5 (anglais) ou 2-8418-0981-1 (français).

Objectives:

Introduce the tools and the mathematical language necessary to the theoretical study of computer science.

Learn the manipulation techniques of fundamental mathematical concepts.

Content:

The course will follow chapters 3-8 of the book, both in class and in small workgroups.

The exercise sessions will consist among others in a personal reading of selected sections of the book.

1. Integer functions
2. Number Theory and Cryptography
3. Binomial Identities
4. Special Numbers
5. Generating Functions
6. Discrete Probability.

Matière examinée / subjects examined		Informatique théorique I,II			
Session	ETE AUT	Coefficient / Crédits ECTS	4	Forme de l'examen / Form of examination	Écrit

Titre	Introduction aux systèmes informatiques
Title	Introduction to computing systems

Enseignant(s) / Instructor(s)	Sanchez Eduardo: IN	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Informatique (2005-2006, Bachelor semestre 1)	Cours: 2 Exercice: 1		
Systèmes de communication (2005-2006, Bachelor semestre 1)	Cours: 2 Exercice: 1		

Objectifs:

Le but est d'établir les fondations de l'informatique, afin de mieux préparer les étudiants aux cours d'approfondissements ultérieurs. Les systèmes informatiques seront présentés comme une hiérarchie des machines virtuelles, dont les différents rôles seront décrits. La structure de base des ordinateurs sera expliquée, en montrant comment une instruction est exécutée et comment les différents types de données sont représentés. Une introduction sera donnée également aux systèmes d'exploitation ainsi qu'aux différents outils et applications de développement du logiciel (compilateur, linker, loader, etc).

Contenu:

1. Introduction.
2. Histoire de l'informatique.
3. Niveaux d'abstraction.
4. Langages de haut niveau.
5. Représentation de l'information : systèmes de numération.
6. Représentation de l'information : nombres entiers et réels.
7. Représentation de l'information non numérique.
8. Organisation de base d'une machine de von Neumann.
9. Langages machine.
10. Traduction des langages.
11. Systèmes d'exploitation.
12. Systèmes logiques : algèbre booléenne.
13. Systèmes logiques : technologie.
14. Test.

Préparation pour:

Systèmes logiques, Architecture des ordinateurs, Programmation, Compiler construction, Systèmes d'exploitation

Forme d'enseignement:

Ex cathedra et exercices

Bibliographie:

Cours polycopié
J. S. Warford, Computer Systems, Jones and Bartlett Publishers, 1999

Objectives:

The goal is to establish the foundations of informatics, in order to better prepare the students for the more in-depth futur courses. Computing systems will be presented as a hierarchy of virtual machines, all of which will be described. The basic structure of computers will be explained, by showing how an instruction is performed and how different data types are represented. An introduction will be also given to operating systems, and to various tools and applications for software development (compiler, linker, loader, etc).

Content:

1. Introduction.
2. History of the computer.
3. Levels of abstraction.
4. High-order languages.
5. Information representation : numerical systems.
6. Information representation : integer and floating-point numbers.
7. Representation of nonnumeric data.
8. Basic organization of a von Neumann machine.
9. Assembly language.
10. Language translation principles.
11. Operating systems.
12. Digital systems : Boolean algebra.
13. Digital systems : technological aspects.
14. Test.

Form of teaching:

Ex cathedra and exercises

Matière examinée / subjects examined		Introduction aux systèmes informatiques			
Session	PRI	Coefficient / Crédits ECTS	2	Forme de l'examen / Form of examination	Ctrl continu

Titre	Physique générale I
Title	General physics I

Enseignant(s) / Instructor(s)	Meylan Georges: PH	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Informatique (2005-2006, Bachelor semestre 2)	Cours: 4 Exercice: 2		

Objectifs:

Connaître et schématiser les phénomènes physiques. Comprendre et savoir utiliser les lois, formulées en termes mathématiques, qui permettent de décrire et prédire ces phénomènes. Applications aux phénomènes naturels et aux domaines techniques.

Contenu:

MECANIQUE

Introduction : Ordres de grandeur. Analyse dimensionnelle.
Cinématique : Référentiels. Trajectoire. Vitesse. Accélération. Systèmes de coordonnées cartésiennes, cylindriques et sphériques.
Dynamique du point matériel : Quantité de mouvement. Lois de Newton. Forces fondamentales, empiriques et de liaison. Mouvement oscillatoire. Moment cinétique. Mouvement de force centrale.
Travail, puissance et énergie : Energies cinétique, potentielle et mécanique. Lois de conservation. Mouvements gravitationnels.
Dynamique des systèmes : Centre de masse. Moment cinétique. Energie. Solide rigide. Gyroscope. Moments d'inertie.
Changements de référentiels : Invariance galiléenne. Accélération d'entraînement et de Coriolis. Dynamique dans les référentiels non inertiels.
Chocs et réactions
Relativité restreinte : Transformation de Lorentz. Quantité de mouvements et énergie relativistes.

THERMODYNAMIQUE

Equilibre : Pression, température, énergie interne.
Echanges d'énergie : Travail et chaleur. Premier principe.
Entropie : Deuxième principe. Cycles. Cycle de Carnot.

Prérequis:

Progressivement Analyse I

Préparation pour:

Physique II

Forme d'enseignement:

Ex cathedra avec expériences en classe

Forme du contrôle:

Test payant facultatif

Bibliographie:

Polycopié

Objectives:

To know and schematize physical phenomena. To understand and apply the laws of physics, formulated in mathematical terms, in order to describe and predict these phenomena. Applications to natural phenomena and technical problems.

Content:

MECHANICS

Introduction: Orders of magnitude. Dimensional analysis.
Kinematics: Referentials Trajectory. Velocity. Acceleration. Cartesian, cylindrical and spherical coordinate systems.
Dynamics of particle: Momentum. Newton's laws. Fundamental forces. Empirical and binding forces. Oscillation motion. Angular momentum. Central force motion.
Work, power, and energy: Kinetic potential and mechanical energies. Conservation laws. Gravitational motions.
Dynamics of systems: Centre of mass. Angular momentum. Energy. Rigid solid. Gyroscope. Moments of inertia.
Changes of referentials: Galilean invariance. Dragging and Coriolis accelerations. Dynamics in non-inertial referentials.
Shocks and reactions
Special relativity: Lorentz transformation. Relativistic momentum and energy.

THERMODYNAMICS

Equilibrium: Pressure, temperature, internal energy.
Energy exchanges: Work and heat. First principle.
Entropy: Second principle. Cycles. Cycle of Carnot.

Required prior knowledge:

Progressively Analysis I

Prerequisite for:

Physics II

Form of teaching:

Ex cathedra with demonstrations in class

Form of examination:

Optional test

Matière examinée / subjects examined		Physique générale I			
Session	ETE AUT	Coefficient / Crédits ECTS	3	Forme de l'examen / Form of examination	Ecrit

Titre	Programmation I
Title	Programming I

Enseignant(s) / Instructor(s)	Sam Jamila: IN	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc / filière /orient	Type
Informatique (2005-2006, Bachelor semestre 1)	Cours:2 Exercice:2 TP:2		obl

Objectifs:

L'objectif de ce cours est de de permettre à l'étudiant de :

- se familiariser avec un environnement informatique (station de travail sous UNIX),
- d'aborder les notions de base de l'informatique logicielle et de l'algorithmique,
- Développer une compétence en programmation et se familiariser avec des concepts de base de la programmation orientée objet (langage JAVA).

Contenu:

Introduction à l'environnement UNIX (connection, multi-fenêtrage, édition de textes, email, ...), éléments de base sur le fonctionnement d'un système informatique et prise en main d'un environnement de programmation (éditeur, compilateur, ...).

Initiation à la programmation (langage JAVA) : variables, expressions, structures de contrôle, modularisation, entrées-sorties, ...

Introduction à la programmation objet (langage JAVA) : Objets, classes, méthodes, encapsulation ...

Présentation informelle de l'algorithmique (exemples, présentation/implantation d'algorithmes connus).

Mise en pratique sur des exemples concrets : les concepts théoriques introduits lors des cours magistraux seront mis en pratique dans le cadre d'exercices sur machines.

Préparation pour:

Programmation II

Forme d'enseignement:

Ex cathedra, travaux pratiques sur ordinateur

Bibliographie:

Polycopié des notes de cours ; ouvrages(s) conseillé(s)indiqué(s) en début de semestre

Objectives:

At the end of this course, the student will have mastered the fundamental aspects of programming and object-oriented programming (using the Java language).

The course will also give an introduction to basic algorithmic concepts.

Content:

Introduction to the Unix development environment

Basics of programming (using Java): variables, expressions, control structures, modularisation, ...

Basics of object-oriented programming (using Java): objects, classes, methods, encapsulation, abstraction, inheritance ...

Introduction to some algorithmic key concepts through the presentation of examples and the implementation of known algorithms

The course topics will be extensively illustrated through practical exercises.

Prerequisite for:

Programming II

Form of teaching:

Ex cathedra, practical work on computer

Matière examinée / subjects examined		Programmation I,II			
Session	ETE	Coefficient / Crédits ECTS	4	Forme de l'examen / Form of examination	Ctrl continu

Titre	Programmation II
Title	Programming II

Enseignant(s) / Instructor(s)	Schinz Michel: IN	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc / filière /orient	Type
Informatique (2005-2006, Bachelor semestre 2)	Cours:2 TP:2		obl

Objectifs:

- Se familiariser à la programmation en Java et à l'approche objet.
- Apprendre les bases de la gestion de projet informatique.

Contenu:

- Programmation en Java : bibliothèques standard, entrées et sorties, exceptions, interfaces graphiques, événements, structures de données avancées, documentation (javadoc)...
- Programmation orientée objet : classes et instances, état et comportement, encapsulation, héritage, polymorphisme, généricité, typage dynamique...
- Gestion de projet informatique : conception modulaire, gestion de versions, méthodes logicielles, cycle de vie logicielle...
- Exercices et mini-projets avec JBuilder, un environnement de développement intégré pour Java.

Prérequis:

Programmation I

Préparation pour:

Programmation III

Forme d'enseignement:

Ex cathedra + exercices

Objectives:

- Become familiar with Java programming and object orientation.
- Learn the basics of software project management.

Content:

- Java programming: standard libraries, input and output, exceptions, graphical user interfaces, events, advanced data structures, documentation (javadoc), etc.
- Object-oriented programming: classes and instances, state and behavior, encapsulation, inheritance, polymorphism, genericity, dynamic binding, etc.
- Software project management: modular design, version management, software methods, software life cycle, etc.
- Exercises and mini-projects with JBuilder, an integrated development environment for Java.

Form of teaching:

ex cathedra and exercises

Matière examinée / subjects examined		Programmation I,II			
Session	ETE	Coefficient / Crédits ECTS	4	Forme de l'examen / Form of examination	Ctrl continu

Titre	Systèmes logiques
Title	Logic systems

Enseignant(s) / Instructor(s)	Sanchez Eduardo: IN	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Informatique (2005-2006, Bachelor semestre 2)	Cours: 2 TP: 2 TP: 2		
Systèmes de communication (2005-2006, Bachelor semestre 2)	Cours: 2 TP: 2 TP: 2		

Objectifs:

Le but est de familiariser l'étudiant avec les composants matériels logiques et numériques des systèmes de traitement de l'information: portes, verrous, bascules, registres, compteurs, circuits arithmétiques, circuits programmables (PAL, PLA, ROM). De lui enseigner l'usage des modes de représentation des systèmes combinatoires et séquentiels: algèbre de Boole, tables de vérité, diagrammes de décision binaire, tables d'états, graphes des états. De lui apprendre des méthodes de synthèse et de simplification des systèmes combinatoires et séquentiels. D'étudier enfin la représentation binaire des nombres et les opérations arithmétiques binaires.

Contenu:

1. Introduction.
2. Implémentation des fonctions logiques.
3. Systèmes combinatoires à deux niveaux.
4. Systèmes combinatoires multiniveaux.
5. Systèmes combinatoires programmables. PLA, PAL, ROM.
6. Représentation binaire des nombres.
7. Systèmes séquentiels.
8. Méthodes de représentation.
9. Compteurs synchrones et asynchrones.
10. Méthodes de synthèse d'un système séquentiel.
11. Systèmes séquentiels programmables.
12. Structure d'un processeur : unité de contrôle et unité de traitement.
13. Test théorique.
14. Test pratique.

Préparation pour:

Architecture des ordinateurs

Forme d'enseignement:

Cours-laboratoire intégré

Bibliographie:

Cours polycopié
J. Wakerly, Digital design, Prentice Hall, 2001

Objectives:

The goal is to familiarize the student with logic and digital hardware components of computing systems: gates, flip-flops, registers, counters, arithmetic circuits, programmable circuits (PAL, PLA, ROM). To teach the student how to represent combinational and sequential systems: Boolean algebra, truth tables, state graphs. To teach the methods of synthesis and simplification of combinational and sequential systems. Finally, to study the binary number notation and the binary arithmetic operations.

Content:

1. Introduction.
2. Implementation of logic functions.
3. Two-level combinational systems.
4. Multiple-level combinational systems.
5. Programmable combinational systems.
6. Binary representation of numbers.
7. Sequential systems.
8. Representation methods.
9. Synchronous and asynchronous counters.
10. Synthesis of sequential systems.
11. Programmable sequential systems.
12. Processor structure: control unit and datapath unit.
13. Theoretical test.
14. Practical test.

Matière examinée / subjects examined		Systèmes logiques			
Session	ETE	Coefficient / Crédits ECTS	2	Forme de l'examen / Form of examination	Ctrl continu



ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

SECTION D'INFORMATIQUE

Cycle Bachelor

(2^{ème} année et 3^{ème} année)

2005 / 2006

Titre	Algorithmique
Title	Algorithmics

Enseignant(s) / Instructor(s)	Shokrollahi Mohammad Amin: MA		Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours	
Informatique (2005-2006, Bachelor semestre 4)	Cours: 4 Exercice: 2			
Passerelle HES - IN (2005-2006, Bachelor semestre 4)	Cours: 4 Exercice: 2		opt	
Systèmes de communication (2005-2006, Bachelor semestre 4)	Cours: 4 Exercice: 2			
Passerelle HES - SC (2005-2006, Bachelor semestre 4)	Cours: 4 Exercice: 2		opt	

Objectifs:

Connaître et savoir utiliser les notions de base des mathématiques théoriques et pratiques. Ce cours permettra au étudiants de résoudre des problèmes aux sciences de l'ingénieur et notamment en informatique.

Lectures en anglais. Support de cours et exercices en français.

Contenu:

Récurrance Mathématique

- Bases mathématiques, compter des régions, problème de coloration, formule d'Euler, codes de Gray, chemins d'arrêtes disjoints.

Analyse d'algorithmes

- Notation O, complexité en temps et espace, relations de récurrence

Structures de données

- Tableaux, listes chaînées, arborescences, monceaux, arbres AVL, tables de hachage, graphes

Planifier des algorithmes par induction

- Evaluer des polynômes, le problème de la vedette, algorithmes du type « diviser pour régner », programmation dynamique

Algorithmes gloutons

Tri et recherche

- Tri par fusion, tri panier, Quicksort, Heapsort, recherche dichotomique, recherché par interpolation, statistiques d'ordre

Algorithmes de graphes et structures de données

- Traverser des graphes, plus court chemin, arbres couvrants, fermeture transitive, décompositions, couplages, flux dans un réseau

Complexité

- Réductions polynomiales, NP-complétude.

Forme d'enseignement:

Cours ex cathedra, exercices en salle

Bibliographie:

Udi Manber, Addison Wesley publisher :

Introduction to Algorithms : A creative approach, 1989.

Cormen, Leiserson, Rivest, Stein : Introduction to Algorithms, MIT Press, 2001.

Objectives:

The main objective of this course is to provide the students with theory and practice of the basic concepts and techniques in algorithmics. The course is designed to enable students to solve problems in engineering and computer science.

Lectures in English. Support materials and exercises in French.

Content:

Mathematical Induction

- Mathematical background, counting regions, coloring problem, Euler's formula, Gray codes, edge-disjoint paths

Analysis of Algorithms

- O-notation, time and space complexity, recurrence relations

Data structures

- Arrays, linked lists, trees, heaps, AVL trees, hashing, graphs

Design of algorithms by induction

- Evaluating polynomials, the celebrity problem, divide-and-conquer algorithms, dynamic programming

Greedy Algorithms

Sorting and searching

- Merge sort, Bucket sort, Quicksort, Heapsort, binary search, interpolation search, order statistics

Graphs algorithms and data structures

- Graphs traversals, shortest paths, spanning trees, transitive closure, decompositions, matching, network flows

Complexity

- Polynomial reductions, NP-completeness.

Form of teaching:

Course ex cathedra, exercises in class

Matière examinée / subjects examined		Algorithmique			
Session	ETE	Coefficient / Crédits ECTS	6	Forme de l'examen / Form of examination	Ecrit

Titre	Analyse III
Title	Analysis III

Enseignant(s) / Instructor(s)	Liebendoerfer Christine: MA		Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours	
Informatique (2005-2006, Bachelor semestre 3)	Cours: 3 Exercice: 2			
Passerelle HES - IN (2005-2006, Bachelor semestre 3)	Cours: 3 Exercice: 2			

Objectifs:

Etudier les procédés de calcul différentiel et intégral des fonctions de plusieurs variables associés aux problèmes des sciences de l'ingénieur.

Contenu:

Arcs, intégrales curvilignes; intégrales de surface.
Analyse vectorielle:

- Champs vectoriels. Travail et circulation. Flux.
- Opérateurs rotationnel et divergence.
- Formules de Stokes et de Gauss. Formules de Green.
- Coordonnées cylindriques et sphériques; Laplacien.
- Applications à quelques modèles physiques.

Séries de Fourier.

Prérequis:

Analyse I, II

Forme d'enseignement:

Cours ex cathedra; exercices en salle

Forme du contrôle:

Examen écrit

Bibliographie:

B. Dacorogna et C. Tanteri, Analyse avancée pour ingénieurs, PPUR, 2002.
K. Arbenz et A. Wohlhauser, Compléments d'analyse, PPUR, 1981 (réimpression 1993).
E. Kreyszig, Advanced engineering mathematics (Chapters 8-10), John Wiley & Sons, 1999.

Objectives:

Study the methods of vector differential and vector integral calculus together with their applications in engineering.

Content:

Curves, line integrals; surface integrals.
Vector calculus:

- Vector fields. Work and circulation. Flux.
- Curl and divergence.
- Formulae of Stokes and Gauss. Green's formulae.
- Cylindrical and spherical coordinates; Laplacian.
- Applications to physical models.

Fourier series.

Required prior knowledge:

Analysis I, II

Form of teaching:

Ex cathedra lecture; exercises in the classroom

Form of examination:

Written exam

Matière examinée / subjects examined		Analyse III			
Session	PRI	Coefficient / Crédits ECTS	5	Forme de l'examen / Form of examination	Ecrit

Titre	Analyse numérique
Title	Numerical analysis

Enseignant(s) / Instructor(s)	Burman Erik: MA	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Informatique (2005-2006, Bachelor semestre 4)	Cours: 2 Exercice: 1		
Génie mécanique (2005-2006, Bachelor semestre 4)	Cours: 2 Exercice: 1		

Objectifs:

L'étudiant apprendra à résoudre pratiquement divers problèmes mathématiques susceptibles de se poser aux ingénieurs et aux informaticiens.

Contenu:

- Stabilité, conditionnement et convergence de problèmes numériques.
- Approximation polynomiale par interpolation et moindres carrés.
- Intégration numérique.
- Méthodes directes pour la résolution de systèmes linéaires.
- Méthodes itératives pour systèmes d'équations linéaires et non linéaires.
- Equations différentielles ordinaires.
- Problèmes aux limites monodimensionnels traités par différences finies et éléments finis.
- Introduction à l'utilisation du logiciel MATLAB.

Prérequis:

Analyse. Algèbre linéaire. Programmation

Forme d'enseignement:

Ex cathedra. Exercices en classe et sur ordinateurs

Forme du contrôle:

Examen écrit

Bibliographie:

- A. Quarteroni et F. Saleri, « Scientific Computing with MATLAB », Springer-Verlag Berlin, 2003.
 A. Quarteroni, R. Sacco et F. Saleri, « Méthodes Numériques pour le Calcul Scientifique », Springer-Verlag France, Paris, 2000.

Objectives:

The student will learn how to approximate several mathematical problems such as linear systems, integration and differentiation, ordinary differential equations.

Content:

- Stability, condition number and convergence of numerical methods.
- Polynomial interpolation and least squares approximation.
- Numerical integration.
- Direct methods for the solution of linear and nonlinear systems.
- Numerical approximation of ordinary differential equations.
- Finite difference approximation of 2-point boundary value problems.
- Introduction to MATLAB.

Required prior knowledge:

Analysis. Linear Algebra.

Form of teaching:

Ex cathedra lecture. Classroom exercises and with computers

Form of examination:

Written exam

URLs	1) http://ppur.epfl.ch/livres/2-88074-363-X.html				
Matière examinée / subjects examined	Analyse numérique				
Session	ETE	Coefficient / Crédits ECTS	3	Forme de l'examen / Form of examination	Ecrit

Titre	Architecture des ordinateurs I
Title	Computer architecture I

Enseignant(s) / Instructor(s)	Ienne Paolo: IN	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Informatique (2005-2006, Bachelor semestre 3)	Cours: 2 TP: 2 TP: 2		
Passerelle HES - IN (2005-2006, Bachelor semestre 3)	Cours: 2 TP: 2 TP: 2		opt
Systèmes de communication (2005-2006, Bachelor semestre 3)	Cours: 2 TP: 2 TP: 2		

Objectifs:

Première partie : Initier l'étudiant à la conception d'un système digital complexe, et plus particulièrement à celle d'un processeur, en introduisant à cet effet les composants et les méthodes de synthèse adéquats. Il s'agit d'étudier la méthodologie de synthèse des machines algorithmiques: décomposition en unité de contrôle et unité de traitement, et synthèse de chacune d'elles. Le langage VHDL et des outils de simulation et de synthèse automatiques sont utilisés.

Deuxième partie : Initier l'étudiant à la structure des processeurs modernes et à l'arithmétique des ordinateurs.

Contenu:

- Langage VHDL (I - IV)
- Mémoires et FPGAs
- Simulation et synthèse
- Décomposition en unité de contrôle et unité de traitement
- Processeurs (I - IV) : Introduction aux systèmes programmables, architecture au niveau du répertoire d'instructions, arithmétique des ordinateurs

Prérequis:

Systèmes logiques

Préparation pour:

Architecture des ordinateurs II, Advanced computer architecture

Forme d'enseignement:

Cours et laboratoires

Bibliographie:

Wakerly, Digital Design, 3rd Ed., Prentice Hall, 2000
Patterson and Hennessy, Computer Organization & Design, 3rd Ed., Morgan Kaufmann, 2005

Objectives:

First part: Learn to design a complex digital system, with particular attention to processors. Introduce for that purpose modern design techniques and the necessary components. Study the design methodology of computing machines: partitioning into control unit and execution unit, logic synthesis of both. VHDL is used together with appropriate simulators and synthesis tools.

Second part: Introduction to modern processors and to computer arithmetic.

Content:

- VHDL (I - IV)
- Memories and FPGAs
- Simulation and logic synthesis
- Partitioning into control- and datapath-unit
- Processors (I - IV): Introduction to programmable systems, Instruction Set Architecture, Computer Arithmetics

Required prior knowledge:

Systèmes logiques

Prerequisite for:

Architecture des ordinateurs II, Advanced computer architecture

Form of teaching:

Course and laboratory work

Matière examinée / subjects examined		Architecture des ordinateurs I,II			
Session	ETE	Coefficient / Crédits ECTS	8	Forme de l'examen / Form of examination	Ctrl continu

Titre	Architecture des ordinateurs II
Title	Computer architecture II

Enseignant(s) / Instructor(s)	Ienne Paolo: IN		Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours	
Informatique (2005-2006, Bachelor semestre 4)	Cours: 2 TP: 2 TP: 2			
Passerelle HES - IN (2005-2006, Bachelor semestre 4)	Cours: 2 TP: 2 TP: 2		opt	
Systèmes de communication (2005-2006, Bachelor semestre 6)	Cours: 2 TP: 2 TP: 2		opt	

Objectifs:

Comprendre la structure des processeurs modernes et en étudier l'architecture, en particulier du point de vue de l'implémentation des unités de traitement et de contrôle, de la maximisation de la performance (pipelining, ordonnancement dynamique, processeurs superscalaires et VLIW), ainsi que des techniques d'organisation du système ayant une influence sur les performances de la machine (mémoire cache, mémoire virtuelle, périphériques, etc.). Ces notions seront illustrées par l'étude des processeurs réels. Un processeur MIPS sera réalisé lors des travaux de laboratoire.

Contenu:

- Performance des ordinateurs
- Procédures
- Entrées/sorties, interruptions et exceptions
- Hiérarchies de mémoire : caches et mémoire virtuelle
- Augmenter la performance : pipelines, ordonnancement dynamique, processeurs superscalaires et VLIW
- Etudes des cas choisis parmi des processeurs récents

Prérequis:

Systèmes logiques, Architecture des ordinateurs I

Préparation pour:

Advanced computer architecture

Forme d'enseignement:

Cours et laboratoires

Bibliographie:

Wakerly, Digital Design, 3rd Ed., Prentice Hall, 2000
Patterson and Hennessy, Computer Organization & Design, 3rd Ed., Morgan Kaufmann, 2005

Objectives:

Understand the structure of modern processors. Study the architecture primarily under the perspective of the datapath- and control-unit design, of the performance enhancement (pipelining, dynamic scheduling, superscalar, VLIW), and of the system organization choices which impact performance (caches, virtual memory, peripherals, etc.). The general notions will be illustrated with real processor examples. A MIPS processor will be designed during the practical sessions.

Content:

- Computer performance
- Procedures
- Inputs/outputs, interrupts, and exceptions
- Memory hierarchy: caches and virtual memory
- Increasing performance: pipelining, dynamic scheduling, superscalar, and VLIW processors
- Case studies selected among recent processors

Required prior knowledge:

Systèmes logiques, Architecture des ordinateurs I

Prerequisite for:

Advanced computer architecture

Form of teaching:

Course and laboratory work

Matière examinée / subjects examined		Architecture des ordinateurs I,II			
Session	ETE	Coefficient / Crédits ECTS	8	Forme de l'examen / Form of examination	Ctrl continu

Titre	Bases de données
Title	Databases

Enseignant(s) / Instructor(s)	Spaccapietra Stefano: IN	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Informatique (2005-2006, Bachelor semestre 4)	Cours: 2 Exercice: 2 Projet: 2		

Objectifs:

Ce cours permet aux étudiants d'acquérir un profil de spécialiste des bases de données (BD). Il forme aux tâches de conception, mise en oeuvre, utilisation et gestion de BD relationnelles:

- exprimer les besoins en information des applications de manière simple et rigoureuse,
- concevoir une BD avec une démarche d'ingénieur,
- implanter une BD sur un système de gestion de bases de données (SGBD) relationnel,
- utiliser les BD au travers des langages de manipulation offerts par les SGBD classiques.
- connaître les principes du fonctionnement interne d'un SGBD.
- maîtriser les facteurs d'optimisation des performances.

Contenu:

1. L'approche base de données

- Nature et objectifs de l'approche; architecture d'un SGBD;
- Cycle de vie d'une base de données.

2. Conception d'une base de données

- Le formalisme conceptuel (objets, liens et propriétés);
- Règles de vérification, de validation, de transformation.

3. Bases de données relationnelles

- Le modèle relationnel et ses règles;
- Les bases théoriques des langages relationnels: algèbre relationnelle, calculs relationnels;
- Langages utilisateurs: SQL, QUEL, QBE;
- Passage du schéma conceptuel au schéma logique.

4. Fonctionnement d'un SGBD

- Dictionnaires de données et gestion du schéma.
- Personnalisation et confidentialité.
- Performances du moteur relationnel: optimisation du traitement des requêtes.
- Performances de stockage et d'accès: fichiers aléatoires dynamiques, B-arbres, k-d arbres.
- Partage des données et accès concurrents. Fiabilité.

5. Pratique d'un SGBD

- Réalisation d'un projet de mise en place et utilisation d'une base de données sur ORACLE, via SQL,

Préparation pour:

Advanced databases

Forme d'enseignement:

Ex cathedra; accompagné d'exercices en classe, de pratique sur ordinateur et de la réalisation d'un projet en équipe.

Bibliographie:

copie des transparents, liste de livres recommandés.

Objectives:

This course lets students acquire a database (DB) specialist profile. Students will learn how to design, install, use and manage a relational DB, including how to:

- Express application information requirements in a simple and rigorous way,
- Design a DB with an engineering approach,
- Install a database on a relational database management system (DBMS),
- Use a DB through the associated manipulation languages.
- Understand how a DBMS performs its work.
- Monitor performances for DBMS applications.

Content:

1. The database approach

- Nature and goals of the approach. Architecture of a DBMS.
- Life cycle of a database.

2. Database design

- A conceptual formalism (objects, links and properties).
- Verification, validation, and transformation rules.

3. Relational databases

- The relational model and its rules.
- Relational algebra, relational calculus.
- User oriented languages: SQL, QUEL, QBE.
- Implementation of a conceptual entity-relationship description on a relational DBMS.

4. DBMS operation

- Data Dictionaries and schema management.
- Supporting users, point of views and data privacy.
- Query processing optimization.
- File structures: dynamic hashing, B-trees, k-d trees.
- Data sharing and concurrent access management
- Recovery techniques for data and application security

5. Practice

- Project development including definition and use of a relational database with ORACLE and SQL.

Prerequisite for:

Advanced databases

Form of teaching:

Ex cathedra; including exercises in class, practice on computer and a group project

Matière examinée / subjects examined		Bases de données			
Session	ETE	Coefficient / Crédits ECTS	5	Forme de l'examen / Form of examination	Ecrit

Titre	Compiler construction
Title	Compiler construction

Enseignant(s) / Instructor(s)	Odersky Martin: IN	Langue / Language	EN
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Informatique (2005-2006, Bachelor semestre 5)	Cours: 2 Exercice: 2 TP: 2		
Systèmes de communication (2005-2006, Bachelor semestre 5)	Cours: 2 Exercice: 2 TP: 2		opt

Objectifs:

Le cours a pour but d'apprendre les aspects fondamentaux de l'analyse des langages informatiques et les rendre applicables. A la fin du cours, l'étudiant devrait :

- Etre capable de définir la syntaxe formelle des langages informatiques
- Etre capable de définir le sens des langages informatiques à travers des interprètes
- Connaître la structure interne et l'implémentation de simples compilateurs
- Etre capable d'écrire un compilateur qui transforme un simple langage de programmation dans le code d'une machine virtuelle
- Connaître les structures communes et dessins utilisés dans la construction d'un compilateur
- Connaître les représentations d'exécution d'importantes constructions de programmation

Buts moins tangibles mais néanmoins importants :

- Améliorer la compréhension des langages de programmation
- Comprendre les compromis entre expressivité, simplicité et performance des langages de programmation
- Expérimenter le dessin et l'implémentation d'un projet de logiciel de certaine taille où la théorie est essentielle pour le succès.

Contenu:

1. Overview, source langages, run-time modèles
2. Généralités sur les langages formels
3. Analyse lexicale
4. Analyse syntaxique
5. Résumé syntaxique
6. Analyse sémantique
7. Run-time organisation
8. Génération de code
9. Garbage collection

Préparation pour:

Advanced compiler construction

Forme d'enseignement:

Ex cathedra, exercices et projets en classe

Bibliographie:

Andrew W. Appel, Modern compiler implementation in Java, Addison-Wesley 1997

Objectives:

The course aims to teach the fundamental aspects of analysing computerlanguages and mapping them into executable form. At the end of thecourse, the student should :

- be able to define the formal syntax of computer languages
- be able to define the meaning of computer languages through interpreters
- know the internal structure and implementation of simple compilers
- be able to write a compiler that maps a simple programming language into the code of a virtual machine
- know common frameworks and design patterns used in compiler construction
- know run-time representations of important programming constructs

Some less tangible, but nevertheless important goals are :

- Improving the understanding of programming languages
- Understanding trade-offs between expressiveness, simplicity, and performance of programming languages,
- Experience the design and implementation of a sizable softwareproject where theory is essential for success.

Content:

1. Overview, source languages and run-time models
2. Review of formal languages
3. Lexical analysis
4. Syntactic analysis
5. Abstract syntax
6. Semantic analysis
7. Run-time organisation
8. Code generation
9. Garbage collection

Prerequisite for:

Advanced compiler construction

Form of teaching:

Ex cathedra, exercises and project in class

Matière examinée / subjects examined		Compiler construction			
Session	PRI	Coefficient / Crédits ECTS	6	Forme de l'examen / Form of examination	Ctrl continu

Titre	Computer graphics
Title	Computer graphics

Enseignant(s) / Instructor(s)	Thalmann Daniel: IN	Langue / Language	EN
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Informatique (2005-2006, Bachelor semestre 5)	Cours: 2 TP: 1		
Génie électrique et électronique (2005-2006, Master semestre 1)	Cours: 2 TP: 1	2	opt
Systèmes de communication (2005-2006, Bachelor semestre 5)	Cours: 2 TP: 1		opt

Objectifs:

Le cours s'adresse à tous les futurs ingénieurs qui devront un jour visualiser graphiquement des objets, des mécanismes, des circuits, des constructions, des matériaux, des phénomènes physiques, chimiques, biomédicaux, électriques, météorologiques etc... Le cours les concepts et les méthodes de base pour modéliser des objets graphiques, les transformer et leur donner des aspects réalistes. Il montre aussi comment on peut tenir compte de l'évolution des formes au cours du temps et explique les principes de la Réalité Virtuelle. A la fin du cours, les étudiants seront capables de réaliser des logiciels graphiques et d'animation sur une station graphique.

Contenu:

1. INTRODUCTION. Historique, matériel graphique, modèles graphiques, transformations visuelles, transformations d'images
2. MODELISATION GEOMETRIQUE. Courbes et surfaces paramétriques, balayages, surfaces implicites
3. RENDU REALISTE. Couleur, visibilité des surfaces, lumière synthétique, transparence simple, lancer de rayons, texture, phénomènes naturels
4. ANIMATION PAR ORDINATEUR. Principes de base, animation par dessins -clés, métamorphoses, animation procédurale, animation de corps articulés, cinématique inverse
5. REALITE VIRTUELLE. Equipements de réalité virtuelle, systèmes de réalité virtuelle

Préparation pour:

Advanced Computer Graphics, Virtual Reality

Forme d'enseignement:

Ex cathedra, films, démos

Forme du contrôle:

avec contrôle continu

Bibliographie:

Notes de cours

Objectives:

This course is dedicated to future engineers who will have someday to visualize graphically objects, mechanisms, circuits, buildings, materials, physical, chemical, biomedical, electric, or meteorological phenomena etc. The course will explain the basic concepts and methods to model graphical objects, transform them and give them realistic aspects. It will also show how take into account the evolution of shapes over time and explain the principles of Virtual Reality. At the end of the course, students will be able to develop graphical and animation software on a graphics workstation.

Content:

1. INTRODUCTION. Historical background, graphics hardware, graphical models, visual transformations, image transformations
2. GEOMETRIC MODELLING. Parametric curves and surfaces, swept surfaces, implicit surfaces
3. REALISM. Color, surface visibility, synthetic light, simple transparency, ray-tracing, texture
4. COMPUTER ANIMATION. Basic principles, key-frame animation, morphing, procedural animation, animation of articulated bodies, inverse kinematics
5. VIRTUAL REALITY. Virtual reality devices, Virtual Reality systems

Prerequisite for:

Advanced Computer Graphics, Virtual Reality

Form of teaching:

Ex cathedra, films, demonstration

Form of examination:

with continuous control

Matière examinée / subjects examined		Computer graphics			
Session	PRI	Coefficient / Crédits ECTS	3	Forme de l'examen / Form of examination	Ecrit

Titre	Concurrence
Title	Concurrence

Enseignant(s) / Instructor(s)	Sandoz Alain: SC	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Informatique (2005-2006, Bachelor semestre 5)	Cours: 2 Exercice: 1		
Systèmes de communication (2005-2006, Bachelor semestre 5)	Cours: 2 Exercice: 1		

Objectifs:

L'étudiant apprendra à concevoir un programme concurrent. Il comprendra également le rôle et le fonctionnement d'un noyau de système concurrent. Il pratiquera ces notions sur un environnement de programmation courant.

Contenu:

Programmation concurrente
 Notion de processus
 Noyau d'un système concurrent
 Exclusion mutuelle et synchronisation
 Événements, sémaphores
 Moniteurs, aspects concurrents du langage Java
 Rendez-vous, aspects concurrents du langage Ada
 Implémentation d'un noyau, aspects concurrents du langage Modula-2
 Threads POSIX

Prérequis:

Programmation I et II

Forme d'enseignement:

Ex cathedra. Exercices en salle et sur ordinateur

Bibliographie:

Programmation concurrente (PPR) + support de cours au format pdf

Objectives:

The student will learn to design a concurrent program. He/she will also understand the role of the kernel of concurrent system. He / she will practice these notions using a standard programming environment.

Content:

Concurrent programming
 Notion of a process
 Kernel of a concurrent system
 Mutual exclusion and synchronization
 Events and semaphores
 Monitors, concurrency aspects in Java
 Rendez-vous, concurrency aspects in Ada
 Implementation of a kernel, concurrency aspects in Modula-2
 POSIX Threads

Form of teaching:

Ex cathedra. Exercises in class and on the computer

Matière examinée / subjects examined		Concurrence			
Session	PRI	Coefficient / Crédits ECTS	3	Forme de l'examen / Form of examination	Ecrit

Titre	Informatique du temps réel
Title	Real-time systems

Enseignant(s) / Instructor(s)	Decotignie Jean-Dominique: SC	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Informatique (2005-2006, Bachelor semestre 5)	Cours: 2 Exercice: 1		

Objectifs:

A l'issue du cours, l'étudiant aura acquis les connaissances principales liées à la conception et la réalisation des systèmes temps réel. Les différentes notions seront illustrées par des exercices et des laboratoires.

Contenu:

1. Introduction sur l'informatique du temps-réel et ses particularités
2. Modélisation des systèmes temps-réel - contexte, types
3. Modélisation asynchrone du comportement logique - Réseaux de Petri
4. Modélisation synchrone - GRAFCET (liens avec les langages synchrones)
5. Programmation des systèmes temps-réels - types de programmation (polling, par interruption, par états, exécutifs cycliques, coroutines, tâches)
6. Noyaux et systèmes d'exploitation temps-réel - problèmes, principes, mécanismes (tâches synchrones et asynchrones, synchronisation des tâches, gestion du temps et des événements)
7. Ordonnancement - problèmes, contraintes, nomenclature
8. Ordonnancement à priorités statiques (Rate Monotonic) et selon les échéances (EDF)
9. Ordonnancement en tenant compte des ressources, des relations de précédence et des surcharges
10. Ordonnancement de tâches multimedia
11. Evaluation des temps d'exécution

Forme d'enseignement:

Ex cathedra + laboratoires

Bibliographie:

H. Nussbaumer, Informatique industrielle II, PPUR + photocopiés

Objectives:

At the completion of the course, the student will have mastered the main topics concerning the design and programming of real-time systems. The course topics will be illustrated through exercises and a practical case study.

Content:

1. Introduction - Real-time systems and their characteristics
2. Model ling real-time systems - context and types
3. Asynchronous models of logical behavior - Petri nets
4. Synchronous models - GRAFCET (link with synchronous languages)
5. Programming real-time systems (polling, cyclic executives, co-routines, state based programming)
6. Real-time kernels and operating systems - problems, principles, mechanisms (synchronous and sporadic tasks, synchronization, event and time management)
7. Scheduling - problem, constraints, taxonomy
8. Fixed priority and deadline oriented scheduling
9. Scheduling in presence of shared resources, precedence constraints and overloads
10. Scheduling of continuous media tasks
11. Evaluation of worst case execution times

Form of teaching:

Ex cathedra + laboratory work

Matière examinée / subjects examined		Informatique du temps réel			
Session	PRI	Coefficient / Crédits ECTS	3	Forme de l'examen / Form of examination	Ecrit

Titre	Intelligence artificielle
Title	Artificial intelligence

Enseignant(s) / Instructor(s)	Faltings Boi: IN	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Informatique (2005-2006, Bachelor semestre 6)	Cours: 2 Projet: 2		
Systèmes de communication (2005-2006, Bachelor semestre 6)	Cours: 2 Projet: 2		opt

Objectifs:

Connaitre les principales techniques pour la réalisation de systèmes à base de connaissances et des agents intelligents.

Contenu:

1. Notions de base: logique des prédicats, inférence et démonstration automatique des théorèmes
2. Programmation symbolique, en particulier en LISP
3. Algorithmes de recherche, moteurs d'inférence, systèmes experts
4. Diagnostic: par raisonnement incertain, par système expert, et par modèles
5. Raisonnement avec des données incertaines: logique floue, inférence Bayésienne
6. Satisfaction de contraintes: définition, consistance et principaux théorèmes, heuristiques de recherche, propagation locale, raisonnement temporel et spatial
7. Planification automatique: modélisation, planification linéaire et non-linéaire
8. Apprentissage automatique: induction d'arbres de décision et de règles, algorithmes génétiques, explanation-based

Prérequis:

Programmation IV

Préparation pour:

Intelligent Agents

Forme d'enseignement:

Ex cathedra, travaux pratiques sur ordinateur

Bibliographie:

Polycopié: Intelligence Artificielle
 Winston & Horn: LISP, Addison Wesley
 Russel & Norvig: Artificial Intelligence: A Modern approach, Prentice Hall

Objectives:

Basic principles for implementing knowledge systems and intelligent agents

Content:

1. Basics: predicate logic, inference and theorem proving
2. Symbolic programming, in particular LISP
3. Search algorithms, inference engines, expert systems
4. Diagnosis: using uncertainty, rule systems, and model-based reasoning
5. Reasoning with uncertain information: fuzzy logic, Bayesian networks
6. Constraint satisfaction: definitions, consistency, search heuristics, local propagation, theoretical limits and complexity
7. Planning: modeling, linear and non-linear planning
8. Machine learning: learning from examples, learning decision trees and rules, genetic algorithms, explanation-based learning, case-based reasoning

Required prior knowledge:

Programmation IV

Form of teaching:

Ex cathedra, practical work on computer

Matière examinée / subjects examined		Intelligence artificielle			
Session	ETE	Coefficient / Crédits ECTS	4	Forme de l'examen / Form of examination	Ctrl continu

Titre	Introduction au marketing et à la finance
Title	Introduction to marketing and finance

Enseignant(s) / Instructor(s)	Schwab Jean-Marc: , Wegmann Alain: SC	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Informatique (2005-2006, Bachelor semestre 6)	Cours: 2		
Systèmes de communication (2005-2006, Bachelor semestre 6)	Cours: 2		opt

Objectifs:

Ce cours présente le processus conduisant de la définition du marché d'une entreprise, au développement de ses stratégies marketing et technologique et à l'implémentation de celles-ci. Le cours introduit ensuite comment, à partir des plans commerciaux définis dans la première partie, une entreprise peut être créée ainsi que les différents mécanismes de financement possible. Le but de ce cours est multiple :

- sensibiliser les ingénieurs à leur rôle dans la compétitivité de l'entreprise ;
- montrer comment une entreprise peut être créée et le financement obtenu.

Contenu:

- Marketing et concept de marketing intégré « Business System » & « Business Definition »
- Plan stratégique
- Création d'entreprise
- Financement

Prérequis:

Comptabilité (J.-M Schwab) ou équivalent

Forme d'enseignement:

Ex cathedra

Bibliographie:

Transparents

Objectives:

This course introduces the process leading from business definition, to strategy development and implementation. The course introduces how, from the business plans developed in the first part, a company can be started and how financing can be found. This course has multiple goals:

- to rise the awareness of the engineer regarding his/her role for the enterprise competitiveness;
- to explain how a startup can be created and financing found.

Content:

- Marketing and integrated marketing concept Business system & Business definition
- Strategic business plan
- Business creation
- Financing

Required prior knowledge:

Accounting (J.-M. Schwab) or equivalence

Form of teaching:

Ex cathedra

Matière examinée / subjects examined		Introduction au marketing et à la finance			
Session	ETE	Coefficient / Crédits ECTS	2	Forme de l'examen / Form of examination	Ctrl continu

Titre	Mathématiques discrètes
Title	Discrete mathematics

Enseignant(s) / Instructor(s)	Hêche Jean-François: MA	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Informatique (2005-2006, Bachelor semestre 5)	Cours: 2 Exercice: 1		

Objectifs:

Les étudiants seront familiarisés avec les principaux modèles de la recherche opérationnelle. Ils sauront utiliser les algorithmes de résolution associés et en auront compris les fondements. Par des exemples et des exercices, ils seront entraînés à la modélisation de problèmes de décision rencontrés par l'ingénieur.

Objectives:

To acquaint students with basic operations research models. To enable them to use some of the main algorithms and understand the underlying theory. To train them to model engineering and management decision problems with appropriate exercises and examples.

Contenu:

Programmation linéaire
Modélisation à l'aide de la programmation linéaire.
Géométrie de la programmation linéaire.
Algorithme du simplexe.
Dualité, algorithme dual.
Analyse de sensibilité.
Systèmes d'inégalités linéaires, polyèdres, lemme de Farkas.
Introduction aux méthodes de points intérieurs.

Programmation convexe
Ensembles et fonctions convexes.
Polyèdres, points extrêmes, sommets.

Notions de la théorie des graphes
Connexité, arbres, chaînes, chemins, cycles, circuits.
Matrices d'adjacence et d'incidence.
Problèmes d'optimisation classiques.
Le problème du transbordement.

Applications à la modélisation
Problèmes d'allocation de ressources, de planification, d'ordonnancement, de transport et de distribution.

Prérequis:

Algèbre linéaire, Algorithmique

Préparation pour:

Graphes et réseaux, Combinatoire, Optimisation

Forme d'enseignement:

Cours ex cathedra et exercices en salle et sur ordinateurs

Forme du contrôle:

Branche d'examen (écrit)

Bibliographie:

D. de Werra, J.-F. Hêche, Th. M. Lieblich, Recherche opérationnelle pour l'ingénieur, vol 1, PPUR, 2003.

Content:

Linear programming
Formulating LP models.
Geometry of linear programming.
Simplex algorithm.
Duality, dual simplex method.
Sensitivity analysis.
Linear inequality systems, polyhedra, Farkas lemma.
Introduction to interior points methods.

Convex programming
Convex sets and functions.
Polyhedra, extreme points, vertices.

Elements of graph theory
Connexity, trees, chains, paths, cycles, circuits.
Adjacency and incidence matrices.
Classic optimisation problems.
Transshipment problem.

Modelling applications
Resource allocation, planning and scheduling, transportation and distribution problems.

Required prior knowledge:

Linear Algebra, Algorithmic

Prerequisite for:

Graphs and networks, Combinatoric, Optimisation

Form of teaching:

Ex cathedra lecture and exercises in the classroom (also on the computer)

Form of examination:

Written exam

Matière examinée / subjects examined		Mathématiques discrètes			
Session	PRI	Coefficient / Crédits ECTS	3	Forme de l'examen / Form of examination	Ecrit

Titre	Physique générale II
Title	General physics II

Enseignant(s) / Instructor(s)	Felix Christian: PH	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Informatique (2005-2006, Bachelor semestre 3)	Cours: 4 Exercice: 2		
Passerelle HES - IN (2005-2006, Bachelor semestre 3)	Cours: 4 Exercice: 2		

Objectifs:

Donner à l'étudiant les notions de base nécessaires à la compréhension des phénomènes physiques qu'il rencontrera dans sa vie professionnelle. Il sera capable de prévoir quantitativement les conséquences de ces phénomènes avec les outils théoriques appropriés. Il possédera dans les domaines traités d'une culture générale indispensable à un ingénieur de bon niveau.

Contenu:

Électricité et magnétisme :

Électrostatique, champ électrique, potentiel. Théorème de Gauss, conducteurs, capacités. Courant électriques stationnaires, loi d'Ohm, lois de Kirchhoff. Magnétostatique, induction, courants de Foucault, self induction, induction mutuelle, transformateurs. Circuits électriques simples : RC, LC, RL, RLC. Équations de Maxwell, ondes électromagnétiques.

Phénomènes ondulatoires :

Étude phénoménologique de diverses ondes (acoustiques, élastiques, électromagnétiques). Modélisation de l'onde acoustique. Équation de d'Alembert. Superposition d'ondes, interférences, battements, diffraction, réflexion.

Prérequis:

Physique générale I

Préparation pour:

Forme d'enseignement:

Ex cathedra avec nombreuses expériences de cours et exercices

Forme du contrôle:

Contrôle continu partiel

Bibliographie:

Giancoli, Physique générale, Ed. de Boeck

Objectives:

To give to the student the basic notions necessary to comprehension of the physical phenomena that he will encounter during his professional life. He will be able to envisage quantitatively the consequences of these phenomena with the suitable theoretical tools. He will have in the covered topics the scientific culture required for a well trained engineer.

Content:

Electricity and magnetism

Electrostatics, electric field, potential, Gauss's law, conductors, capacitance. Stationary electric currents, Ohm's law, Kirchhoff's rules. Magnetostatics, induction, Eddy current, self induction, mutual induction, transformers. Simple electric circuits: RC, LC, RL, RLC. Maxwell's equations, electromagnetic waves.

Waves:

Phenomenological overview of different type of waves (acoustic, elastic, electromagnetic). Modeling of an acoustic wave. D'Alembert's equation, wave superposition, interference, beats, diffraction, reflection

Required prior knowledge:

General physics I

Form of teaching:

Ex cathedra with demonstration and exercises

Form of examination:

Continuous control partial

Matière examinée / subjects examined		Physique générale II			
Session	PRI	Coefficient / Crédits ECTS	6	Forme de l'examen / Form of examination	Écrit

Titre	Probabilités et statistique I
Title	Probabilities and statistic I

Enseignant(s) / Instructor(s)	Mountford Thomas: MA	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Informatique (2005-2006, Bachelor semestre 3)	Cours: 2 Exercice: 1		
Sciences et ingénierie de l'environnement (2005-2006, Bachelor semestre 3)	Cours: 2 Exercice: 1		

Objectifs:

Les notions de probabilité sont importantes en elles-mêmes et servent de base à la théorie des statistiques que l'on traitera dans la deuxième partie du cours. Je voudrais exposer les résultats de base ainsi que donner un aperçu de l'importance quotidienne des idées probabilistes.

Contenu:

- Résultats combinatoires, y compris la forme binomiale.
- Le théorème de Bayes et la probabilité conditionnelle.
- L'indépendance. La formule des probabilités totales.
- Les variables aléatoires. Les lois naturelles et utiles y compris la loi de Poisson, la gaussienne, la binomiale, l'exponentielle.
- L'espérance, la variance, la corrélation et leur signification intuitive.
- La loi des grands nombres.
- Le théorème de limite centrale.

Préparation pour:

Probabilités et statistique II

Forme d'enseignement:

Cours ex cathedra et exercices en classe

Forme du contrôle:

Continu

Bibliographie:

Initiation aux probabilités par S. Ross, PPUR

Objectives:

The notions of probability, besides their intrinsically importance, serve as a base for the theory of statistics for the second part of the course. In addition to introducing the fundamental ideas, I wish to transmit the usefulness of probabilistic ideas in everyday life.

Content:

- Combinatorics, including the binomial formula Bayes theorem and conditional probability, the formula of total probability independence.
- Random variables, natural and much used families of distributions including Poisson, Gaussian, binomial and exponential.
- Expectation, variance and correlation and their intuitive meaning.
- The law of large numbers.
- The central limit theorem.

Prerequisite for:

Probability and statistics II

Form of teaching:

Ex cathedra lecture and exercises

Form of examination:

Constant

Matière examinée / subjects examined	Probabilités et statistique I,II		
Session	ETE AUT	Coefficient / Crédits ECTS	6
Forme de l'examen / Form of examination			Ecrit

Titre	Probabilités et statistique II
Title	Probabilities and statistic II

Enseignant(s) / Instructor(s)	Mountford Thomas: MA	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Informatique (2005-2006, Bachelor semestre 4)	Cours: 2 Exercice: 1		
Sciences et ingénierie de l'environnement (2005-2006, Bachelor semestre 4)	Cours: 2 Exercice: 1		

Objectifs:

Le cours a pour but de sensibiliser les étudiants, d'une part aux faits et résultats de base des statistiques et d'autre part aux limites des techniques présentées et à leurs interprétations.

Contenu:

- Les questions d'échantillonnage, l'échantillon simple et l'échantillon stratifié. Pourquoi l'on emploie les moyens probabilistes ?
- Les estimateurs et leurs propriétés asymptotiques. La théorie asymptotique des estimateurs de maximum vraisemblance.
- Les tests d'hypothèses dont le test z, le test t, le test du khi-deux, la théorie asymptotique à l'arrière plan. Le lemme de Neyman Pearson.
- Les intervalles de confiance et leur lien avec les tests d'hypothèses.
- L'analyse de variance à plusieurs niveaux et la régression simple.

Prérequis:

Probabilités et statistique I

Forme d'enseignement:

Cours ex cathedra et exercices en classe

Forme du contrôle:

Continu

Bibliographie:

Introductory Statistics with R, Peter Dalgaard

Objectives:

The course aims to introduce the students to the basic ideas and techniques but also to stress the interpretations of the results.

Content:

- Sampling: the motivation for probabilistic sampling, what is a simple random sample, what is stratified sampling.
- Estimations and their asymptotic properties, the theory of the maximum likelihood estimations.
- Hypothesis testing, including the z and t tests, the chi-squared test.
- Links between hypothesis tests and intervals of confidence.
- Simple and multiple regression.

Required prior knowledge:

Probability and statistics I

Form of teaching:

Ex cathedra lecture and exercises

Form of examination:

Constant

Matière examinée / subjects examined	Probabilités et statistique I,II		
Session	ETE AUT	Coefficient / Crédits ECTS	6
Forme de l'examen / Form of examination			Ecrit

Titre	Programmation III
Title	Programming III

Enseignant(s) / Instructor(s)	Chappelier Jean-Cédric: IN	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Informatique (2005-2006, Bachelor semestre 3)	Cours: 2 TP: 2		
Passerelle HES - IN (2005-2006, Bachelor semestre 3)	Cours: 2 TP: 2		opt

Objectifs:

L'objectif de ce cours est de développer une compétence de base en programmation orientée système (langages UNIX Shell, C et Perl) et de familiariser les étudiants avec l'utilisation d'une station de travail sous UNIX.

À l'issue de ce cours, les étudiants devraient être à même :

- d'écrire des programmes avancés en C qui utilisent les arguments de ligne de commande, des pointeurs et des structures, manipulent la mémoire et les fichiers,... ;
- d'écrire des scripts systèmes simples en Shell (tcsh) et en Perl ;
- d'utiliser les outils systèmes UNIX élémentaires, aussi bien au niveau utilisateur que programmeur.

Contenu:

Rappel des éléments de base du fonctionnement d'un système informatique et de l'environnement UNIX.

Initiation à la programmation en C, puis en Shell puis en Perl : variables, expressions, structures de contrôle, fonctions, entrées-sorties, expressions régulières, ...

Approfondissement des spécificités de la programmation système rudimentaire : utilisation de la mémoire (pointeurs), gestion des fichiers et autres entrées/sorties.

Les concepts théoriques introduits lors des cours magistraux seront mis en pratique dans le cadre d'exercices sur machine.

Prérequis:

Programmation I et II

Préparation pour:

Programmation IV

Forme d'enseignement:

Ex cathedra, travaux pratiques sur ordinateur

Bibliographie:

Notes de cours ; livre(s) de référence indiqué(s) en début de semestre

Objectives:

This course focuses on the basis of system-oriented programming, using C, UNIX Shell and Perl languages. It aims at introducing the basics of using and programming on a UNIX workstation.

At the end of this course, students should be able to:

- write advanced C programs, with command-line arguments, pointers and structures, memory and file handling;
- write Perl and shell scripts (tcsh);
- use the basic tools of a UNIX system, both at the user and programmer level.

Content:

Basics of UNIX environment [reminder]

Introduction to C, then shell and then Perl languages: variables, expressions, structures, control, functions, basic IO, regular expressions, ...

Basics of system-oriented programming: memory (pointers), file handling, misc. IO.

Theoretical concepts presented during plenary lectures will be studied further on UNIX workstations during practical sessions.

Required prior knowledge:

Programmation I and II

Prerequisite for:

Programmation IV

Form of teaching:

Ex cathedra, practical work on computer

Matière examinée / subjects examined		Programmation III,IV			
Session	ETE	Coefficient / Crédits ECTS	8	Forme de l'examen / Form of examination	Ctrl continu

Titre	Programmation IV
Title	Programming IV

Enseignant(s) / Instructor(s)	Odersky Martin: IN	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Informatique (2005-2006, Bachelor semestre 4)	Cours: 2 TP: 2		
Passerelle HES - IN (2005-2006, Bachelor semestre 4)	Cours: 2 TP: 2		opt

Objectifs:

Comprendre les principes et applications de la programmation déclarative
 Comprendre des modèles fondamentaux de l'exécution des logiciels
 Comprendre et utiliser des méthodes fondamentales de la composition des logiciels
 Comprendre la méta-programmation par la construction interprètes
 Comprendre les concepts de base de la programmation concurrente

Objectives:

To understand the principles and the applications of declarative programming.
 To understand the fundamental models of program execution.
 To understand and use fundamental techniques of software composition.
 To understand meta-programming by building interpreters.
 To understand the basis of concurrent programming.

Contenu:

Introduction au langage Scala
 Expressions et fonctions
 Enregistrements et objets
 Evaluation par réécriture
 Types algébriques
 Polymorphisme
 Stratégies de l'évaluation
 Objets avec état
 Flots et Itérateurs
 Interprètes des langages
 Un interprète pour LISP
 Un interprète pour Prolog
 Unification

Content:

Introduction to the Scala language
 Expressions and functions
 Records and objects
 Evaluation through rewriting
 Algebraic data-types
 Polymorphism
 Evaluation strategies
 Objects with states
 Streams and iterators
 Language interpreters
 A LISP interpreter
 A Prolog interpreter
 Unification

Prérequis:

Programmation I, II, III

Required prior knowledge:

Programmation I, II, III

Forme d'enseignement:

Ex cathedra. Exercices et projets sur ordinateur

Form of teaching:

Ex cathedra. Computer exercises and projects

Bibliographie:

Voir sous URL

URLs	1) http://lampwww.epfl.ch/courses/programmation-iv			
Matière examinée / subjects examined	Programmation III,IV			
Session	ETE	Coefficient / Crédits ECTS	8	Forme de l'examen / Form of examination
				Ctrl continu

Titre	Programmation internet
Title	Internet programming

Enseignant(s) / Instructor(s)	Petitpierre Claude: IN	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Informatique (2005-2006, Bachelor semestre 4)	Cours: 2 TP: 2		

Objectifs:

Le but de ce cours est d'apprendre à utiliser les composants et bibliothèques de Java dans des situations proches de celles rencontrées dans l'industrie.
Il présentera les concepts indiqués ci-dessous dans des architectures réalistes et montrera comment coordonner ces concepts.

Contenu:

J2EE

Cette bibliothèque offre les moyens de construire des applications exécutées sur le Web. Elle permet la gestion de servelettes, d'objets permettant l'accès à des bases de données (session beans, entity beans), de clients, de messages asynchrones, d'accès à distance (RMI), etc.

Architectures de logiciel réparti

L'implémentation de systèmes répartis pose un certain nombre de problèmes particuliers pour lesquels des architectures générales utilisables dans différentes situations seront présentées et réalisées.

Système de développement

Le cours est basé sur l'utilisation d'Eclipse, de JBoss et de modules préparés par l'enseignant. Tous ces programmes font partie du domaine public et peuvent être exécutés sur des laptops. Ils sont toutefois utilisés par l'industrie pour réaliser des projets complexes.

Prérequis:

Programmation I et II

Forme d'enseignement:

Ex cathedra + travaux pratiques

Bibliographie:

Cours polycopiés, documents sur le Web

Objectives:

The goal of this lecture is to learn how to use Java components and libraries in situations close to those encountered in industry. It will present the concepts indicated below within realistic architecture and will show how to coordinate these concepts.

Content:

J2EE

This library offers means to build applications executed on the Web. It allows the management of servlets, of object accessing databases (session beans, entity beans), of clients, of asynchronous messages, of remote accesses (RMI), and so on.

Architectures of distributed software

The implementation of distributed systems raises a number of particular problems for which general architectures usable in various situations will be presented and realized.

Development system

The course is based on the use of Eclipse and JBoss and a module prepared by the teacher. All these programs are public domain and can be executed on laptops. However, they are used in the industry to realize complex projects.

Required prior knowledge:

Programmation I et II

Form of teaching:

Ex cathedra + practical work

Matière examinée / subjects examined		Programmation internet			
Session	ETE	Coefficient / Crédits ECTS	3	Forme de l'examen / Form of examination	Ctrl continu

Titre	Projet de génie logiciel I
Title	Software engineering project I

Enseignant(s) / Instructor(s)	Hulaas Jarle: IN, Petitpierre Claude: IN	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Informatique (2005-2006, Bachelor semestre 5)	TP: 5 TP: 5		

Objectifs:

Maîtriser le développement d'une application logicielle de complexité moyenne. Savoir appliquer une méthode de développement par objets. Vivre l'expérience d'un travail d'équipe.

Contenu:

Réalisation d'un projet logiciel par des groupes d'étudiants (en général au nombre de cinq). Le développement sera basé sur UML. Il consistera à mettre en oeuvre une application qui demande la coordination de plusieurs aspects système : répartition, bases de données, serveurs Web, GUI, multi-tâche, etc.

NOTE 1

Cet enseignement est annuel. Il ne peut pas être fractionné.

NOTE 2

Cet enseignement est dédoublé, un groupe chez Cl. Petitpierre et un groupe chez J. Hulaas. La répartition des étudiants se fera au début du semestre d'hiver pour toute l'année, sur la base d'un algorithme « premier groupe venu, premier groupe servi ».

Prérequis:

Software engineering

Forme d'enseignement:

Projet en équipe

Remarque:

Pour le groupe "Petitpierre" : chaque étudiant est amené à faire un exposé

<http://ltiwww.epfl.ch/~petitp/SoftwareEngineering/index.html>

Pour le groupe "Hulaas" :

http://lglwww.epfl.ch/teaching/software_project/home_page.html

Voir aussi cours "Software engineering"

Bibliographie:

Voir sous "Remarque"

Objectives:

To master the development of a medium-size software application. To be able to apply an object-oriented software development method. To experience working in a team.

Content:

Development of a software application by teams of students (usually five of them). The development will be based on UML. It will consist of developing an application that requires the coordination of several system aspects: distribution, databases, Web servers, GUI, multi-tasking, and so on.

NOTE 1

This class lasts for the whole academic year. It cannot be divided.

NOTE 2

This class is split in two, and also given by Mr. Jarle Hulaas . The students will be divided between the two classes at the beginning of the academic year, on the basis "first group registered, first group enrolled".

Required prior knowledge:

Software engineering

Form of teaching:

Group project

Note:

For the "Petitpierre" group : each student is requested to do a presentation

<http://ltiwww.epfl.ch/~petitp/SoftwareEngineering/index.html>

For the "Hulass" group :

http://lglwww.epfl.ch/teaching/software_project/home_page.html

See also course "Software engineering"

URLs	1) http://lglwww.epfl.ch/teaching/software_project/home_page.html				
URLs	1) http://ltiwww.epfl.ch/~petitp/SoftwareEngineering/index.html				
Matière examinée / subjects examined	Projet de génie logiciel I,II				
Session	ETE	Coefficient / Crédits ECTS	10	Forme de l'examen / Form of examination	Ctrl continu

Titre	Projet de génie logiciel II
Title	Software engineering project II

Enseignant(s) / Instructor(s)	Hulaas Jarle: IN, Petitpierre Claude: IN	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Informatique (2005-2006, Bachelor semestre 6)	TP: 5 TP: 5		

Objectifs:

Maîtriser le développement d'une application logicielle de complexité moyenne. Savoir appliquer une méthode de développement par objets. Vivre l'expérience d'un travail d'équipe.

Contenu:

Réalisation d'un projet logiciel par des groupes d'étudiants (en général au nombre de cinq). Le développement sera basé sur UML. Il consistera à mettre en oeuvre une application qui demande la coordination de plusieurs aspects système : répartition, bases de données, serveurs Web, GUI, multi-tâche, etc.

NOTE 1

Cet enseignement est annuel. Il ne peut pas être fractionné.

NOTE 2

Cet enseignement est dédoublé, un groupe chez Cl. Petitpierre et un groupe chez J. Hulaas. La répartition des étudiants se fera au début du semestre d'hiver pour toute l'année, sur la base d'un algorithme « premier groupe venu, premier groupe servi ».

Prérequis:

Software engineering

Forme d'enseignement:

Projet en équipe

Remarque:

Pour le groupe "Petitpierre" : chaque étudiant est amené à faire un exposé
<http://ltiwww.epfl.ch/~petitp/SoftwareEngineering/index.html>

Pour le groupe "Hulaas" :
http://lglwww.epfl.ch/teaching/software_project/home_page.html

Voir aussi cours "Software engineering"

Bibliographie:

Voir sous "Remarque"

Objectives:

To master the development of a medium-size software application. To be able to apply an object-oriented software development method. To experience working in a team.

Content:

Development of a software application by teams of students (usually five of them). The development will be based on UML. It will consist of developing an application that requires the coordination of several system aspects: distribution, databases, Web servers, GUI, multi-tasking, and so on.

NOTE 1

This class lasts for the whole academic year. It cannot be divided.

NOTE 2

This class is split in two, and also given by Mr. Jarle Hulaas . The students will be divided between the two classes at the beginning of the academic year, on the basis "first group registered, first group enrolled".

Required prior knowledge:

Software engineering

Form of teaching:

Group project

Note:

For the "Petitpierre" group : each student is requested to do a presentation
<http://ltiwww.epfl.ch/~petitp/SoftwareEngineering/index.html>

For the "Hulass" group :
http://lglwww.epfl.ch/teaching/software_project/home_page.html

See also course "Software engineering"

URLs	1) http://ltiwww.epfl.ch/~petitp/SoftwareEngineering/index.html 2) http://lglwww.epfl.ch/teaching/software_project/home_page.html			
Matière examinée / subjects examined	Projet de génie logiciel I,II			
Session	ETE	Coefficient / Crédits ECTS	10	Forme de l'examen / Form of examination
				Ctrl continu

Titre	Recherche opérationnelle
Title	Operations research

Enseignant(s) / Instructor(s)	Hêche Jean-François: MA	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Informatique (2005-2006, Bachelor semestre 6)	Cours: 2 Exercice: 1		

Objectifs:

Les étudiants seront familiarisés avec les principaux modèles de la recherche opérationnelle. Ils sauront utiliser les algorithmes de résolution associés et en auront compris les fondements. Ils auront acquis des notions de modélisation mathématique de problèmes de décision, en particulier en présence d'éléments stochastiques.

Contenu:

Optimisation séquentielle

Programmation dynamique déterministe.
Applications : problème du sac à dos, problèmes de plus courts chemins, problème de renouvellement d'équipement.

Introduction aux processus stochastiques de décision

Programmation dynamique stochastique.
Application à la gestion des stocks.
Chaînes de Markov finies à temps discret et continu.
Propriétés et applications.
Classification des états d'une chaîne de Markov
Discussion du régime transitoire et stationnaire.
Chaînes de Markov absorbantes : temps et probabilités d'absorption

Files d'attente

Processus de Poisson, marches aléatoires.
Processus de naissance et de mort.
Classification des files d'attente simples, notation de Kendall.
Formule de Little.
Files d'attente markoviennes : M/M/1, M/M/s, M/M/infini, M/M/s/K.
File M/G/1, formule de Pollaczek-Khinchin.
Réseaux de Jackson, réseaux à forme produit.

Prérequis:

Mathématiques discrètes, probabilités

Préparation pour:

Ordonnancement et conduite de systèmes, Modèles stochastiques pour les communications, Optimisation

Forme d'enseignement:

Cours ex cathedra et exercices en salle et sur ordinateurs

Bibliographie:

J.-F. Hêche, Th. M. Lieblich, D. de Werra, Recherche opérationnelle pour l'ingénieur, vol 2, PPUR, 2003.

Objectives:

To acquaint students with basic operations research models. To enable them to use some of the main algorithms and understand the underlying theory. To train them to model engineering and management decision problems in a stochastic environment.

Content:

Sequential optimisation

Deterministic dynamic programming.
Applications: knapsack problem, shortest paths problems, machine replacement problem.

Introduction to stochastic decision processes

Stochastic dynamic programming.
Applications in inventory control.
Discrete and continuous time finite Markov chains.
Properties and applications.
Markov chain state classification.
Discussion of transient and stationary modes.
Absorbing Markov chains, hitting times and absorption probabilities

Queuing theory

Poisson processes, random walks.
Birth and death processes.
Classification of simple queuing systems, Kendall's notation Little's formula.
Markovian queues: M/M/1, M/M/s, M/M/∞, M/M/s/K.
M/G/1 queue, Pollaczek-Khinchin's formula.
Jackson queuing networks, product form networks.

Required prior knowledge:

Discrete mathematics, probabilities

Prerequisite for:

Scheduling and System management, Stochastic models for communications, Optimisation

Form of teaching:

Ex cathedra lecture and exercises in the classroom and on computers

URLs	1) http://roso.epfl.ch/cours/optimisation_l/				
Matière examinée / subjects examined	Recherche opérationnelle				
Session	ETE	Coefficient / Crédits ECTS	3	Forme de l'examen / Form of examination	Ecrit

Titre	Réseaux informatiques
Title	Computer networks

Enseignant(s) / Instructor(s)	Grossglauser Matthias: SC	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Informatique (2005-2006, Bachelor semestre 6)	Cours: 2 Exercice: 2		
Systèmes de communication (2005-2006, Bachelor semestre 4)	Cours: 2 Exercice: 2		
Passerelle HES - SC (2005-2006, Bachelor semestre 4)	Cours: 2 Exercice: 2		opt

Objectifs:

Connaître les principes des réseaux TCP/IP. Savoir écrire un programme client ou serveur TCP ou UDP.

Contenu:

The principles of computer networking. Layers, connection oriented versus connectionless operations. Services and Protocols. Architectures.

The connectionless network layer of the Internet. IP v4 and IP v6. ICMP, ARP, packet forwarding versus routing. Multicast IP

The transport layer of the Internet: TCP, UDP.

The domain name system of the Internet.

UNIX networking commands.

Socket programming.

Prérequis:

Initiation au langage de programmation C

Préparation pour:

Computer Networking II

Forme d'enseignement:

Ex cathedra. Travaux pratiques sur ordinateur

Forme du contrôle:

Avec contrôle continu

Objectives:

Know the principles of TCP/IP networks. Be able to write a UDP or TCP server or client program.

Content:

The principles of computer networking. Layers, connection oriented versus connectionless operations. Services and Protocols. Architectures.

The connectionless network layer of the Internet. IP v4 and IP v6. ICMP, ARP, packet forwarding versus routing. Multicast IP

The transport layer of the Internet: TCP, UDP.

The domain name system of the Internet.

UNIX networking commands

Socket programming.

Required prior knowledge:

Initiation to the C programming language

Form of teaching:

Ex cathedra. Practical work on the computer

Form of examination:

With continuous control

Matière examinée / subjects examined		Réseaux informatiques			
Session	ETE	Coefficient / Crédits ECTS	4	Forme de l'examen / Form of examination	Ecrit

Titre	Software engineering
Title	Software engineering

Enseignant(s) / Instructor(s)	Baar Thomas: IN	Langue / Language	EN
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Informatique (2005-2006, Bachelor semestre 5)	Cours: 4		

Objectifs:

Maîtriser une méthode de développement de logiciels par objets.

Contenu:

Résumé: Méthode de développement par objets Fondue (UML), ses modèles et son processus de développement. Eléments de conception de l'interface homme-machine. Documentation d'utilisation du logiciel.

Modèles d'analyse: 1. Modèle des classes du domaine et d'analyse: classe, association, multiplicités, agrégation, généralisation et spécialisation, structuration du modèle des classes. 2. Modèle du contexte du système: acteurs, système, événements. 3. Modèle des opérations du système: pré- et postconditions, schémas d'opération; langage OCL, langage de contraintes sur les objets. 4. Protocole d'interface du système.

Processus d'analyse et vérifications, y compris utilisation de scénarios et cohérence des modèles.

Modèles de conception: 1. Modèle d'interactions: diagrammes de collaborations entre objets, objets et collections d'objets, envoi de messages, enchaînements de messages. 2. Modèle de dépendances entre objets et classes, et leurs caractéristiques. 3. Modèle d'héritage. 4. Modèle des classes de conception.

Processus de conception: Contrôleurs et collaborateurs, décomposition hiérarchique, interface d'utilisateur, architecture client-serveur, héritage versus généralisation et spécialisation, principes de "bonne" conception. Vérifications.

Mappage de la conception vers un langage de programmation: 1. Modèle des classes d'implémentation. 2. Interface de classe: héritage, attributs, méthodes, public versus privé. Mappage de collections. 3. Découplage de classes. 4. Implémentation des méthodes: itérateurs, traitement des erreurs. 4. Implémentation du protocole d'interface du système.

Processus d'implémentation: mappage, performance, vérifications.

Préparation pour:

Projet génie logiciel

Forme d'enseignement:

Ex cathedra. Exercices sur papier

Bibliographie:

Alfred Strohmeier; Overview of the Object-Oriented Technology; EPFL, Switzerland.
Alfred Strohmeier; Fondue Tutorial; EPFL, Switzerland.
Craig Larman; Applying UML and Patterns; Prentice-Hall, 1998.

Objectives:

To master an object-oriented software development method.

Content:

Abstract: The object-oriented development method Fondue (UML), its notations and its development process. Introduction to the design of human-computer interfaces. Users' Documentation.

Analysis Models: 1. Domain and Analysis Class Models: Class, Association, Multiplicities, Aggregation, Generalization and Specialization, Structuring Class Models. 2. System Context Model: actors, system, events. 3. System Operation Model: pre- and postconditions, operation schema; language OCL, the Object Constraint Language. 4. System Interface Protocol.

Analysis process and verifications, including the use of scenarios and consistency between models.

Design Models: 1. Interaction Model: collaboration diagrams, objects and object collections, message sending, message sequencing. 2. Dependency Model: usage dependency and references, other characteristics. 3. Inheritance Model. 4. Design Class Model.

Design process: Controllers and collaborators, hierarchical decomposition, user interface, client-server architecture, inheritance versus generalization-specialization, principles of good design. Checks.

Mapping a design to a programming language: 1. Implementation class model. 2. Class interface: inheritance, attributes, methods, public versus private features, mapping collections. 3. Decoupling classes. 4. Implementing methods: iterators, error handling. 4. Implementing the system interface protocol.

Implementation process: mapping, performance, checks.

Prerequisite for:

Projet génie logiciel

Form of teaching:

Ex cathedra. Exercises on paper

URLs	1) http://glwww.epfl.ch/teaching/software_engineering/home_page				
Matière examinée / subjects examined	Software engineering				
Session	PRI	Coefficient / Crédits ECTS	4	Forme de l'examen / Form of examination	Ctrl continu

Titre	Systèmes d'exploitation
Title	Operating systems

Enseignant(s) / Instructor(s)	Sandoz Alain: SC	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Informatique (2005-2006, Bachelor semestre 6)	Cours: 2 Exercice: 1		opt
Systèmes de communication (2005-2006, Bachelor semestre 6)	Cours: 2 Exercice: 1		opt

Objectifs:

L'étudiant apprendra le rôle, les principes de base et le fonctionnement d'un système d'exploitation.

Contenu:

Introduction aux systèmes d'exploitation

Fonctions d'un système d'exploitation.
Evolution historique des systèmes d'exploitation et terminologie: spooling, multiprogramming, systèmes batch, temps partagé, temps réel. Concept de micro-noyau.

Concepts de Windows NT

Gestion des ressources

Gestion du processeur.
Gestion de la mémoire principale: gestion par zones, gestion par pages (mémoire virtuelle).
Gestion des ressources non préemptibles: le problème de l'interblocage.
Concept de machine virtuelle.

Gestion de l'information

Le système de fichiers, structure logique et organisation physique d'un fichier, contrôle des accès concurrents.
Partage et protection de l'information: matrice des droits, limitation de l'adressage à 1 dimension, adressage segmenté, adressage par capacités.

Prérequis:

Programmation I et II

Forme d'enseignement:

Ex cathedra. Exercices en salle et sur ordinateur

Bibliographie:

Programmation concurrente (PPR) + support de cours en format pdf

Objectives:

The student will learn the role and the basic principles of an operating system, and the way it works.

Content:

Introduction to operating systems

Functions of an operating system.
Historical evolution and terminology: spooling, multiprogramming, batch, time-sharing, real-time. Micro-kernels.

Windows NT concepts

Resource management

Processor management.
Main memory management: contiguous storage allocation, paging (virtual memory).
Management of non-preemptive resources: the deadlock problem.
Virtual machine.

Information management

File systems, logical and physical organisation, concurrency control.
Information sharing and protection: access matrix, limitation of 1 dimensional addressing mechanisms, segmentation, capability.

Form of teaching:

Ex cathedra. Exercises in class and on the computer

Matière examinée / subjects examined		Systèmes d'exploitation			
Session	ETE	Coefficient / Crédits ECTS	4	Forme de l'examen / Form of examination	Ecrit

Titre	Systèmes répartis
Title	Distributed systems

Enseignant(s) / Instructor(s)	Schiper André: SC	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Informatique (2005-2006, Bachelor semestre 6)	Cours: 2 Exercice: 1		

Objectifs:

Les algorithmes répartis constituent l'algorithmique fondamentale de nombreuses applications et systèmes de communication. On peut citer par exemple les applications de réservation, la finance, le contrôle de trafic aérien, ainsi que la gestion des systèmes de communication.

L'objectif de ce cours est de présenter aux étudiants les fondements des algorithmes répartis et de leur apprendre à aborder de manière rigoureuse les problèmes de distribution et leurs solutions.

Contenu:

1. Concepts de base

Etat global, coupe cohérente, horloges logiques, ordonnancement causal, calcul d'état global, propriétés stables, détection de propriétés stables.

2. Tolérance aux défaillances

Critères de cohérence, duplication active, duplication passive, groupes statiques, groupes dynamiques, diffusion totalement ordonnée, diffusion vue-synchrone, consensus, détecteurs de faute, quorums, checkpointing.

3. Transactions réparties

Rappel des propriétés ACID, contrôle de concurrence, atomicité vs durabilité, protocole de validation atomique 2PC et 3PC, réplication de bases de données.

Prérequis:

Systèmes d'exploitation

Forme d'enseignement:

Ex cathedra

Bibliographie:

Notes polycopiées

Objectives:

Distributed algorithms are at the heart of many applications and communication systems. These include banking, reservation and air traffic control applications, as well as network management systems.

The aim of this course is to expose the students to the fundamentals of distributed algorithms and teach them how to approach and reason in a rigorous manner about problems of distribution and their solutions.

Content:

1. Basic concepts

Global state, consistent cut, logical clocks, causal ordering, snapshot algorithm, stable properties, detection of stable properties.

2. Fault-tolerance

Consistency criteria, active replication, primary-backup replication, static groups, dynamic groups, total order broadcast, view-synchronous broadcast, consensus, failure detectors, quorum systems, checkpointing.

3. Distributed transactions

The ACID properties, concurrency control, atomicity vs durability, the 2PC and 3PC atomic commitment protocols, database replication.

Form of teaching:

Ex cathedra

Matière examinée / subjects examined		Systèmes répartis			
Session	ETE	Coefficient / Crédits ECTS	4	Forme de l'examen / Form of examination	Ecrit

Titre	Theoretical computer science III
Title	Theoretical computer science III

Enseignant(s) / Instructor(s)	Henzinger Thomas: IN		Langue / Language	EN
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours	
Informatique (2005-2006, Bachelor semestre 3)	Cours: 4 Exercice: 2			
Passerelle HES - IN (2005-2006, Bachelor semestre 3)	Cours: 4 Exercice: 2		opt	

Objectifs:

Ce cours est une introduction à la « théorie du calcul ».

En bref, le but de ce cours est de fournir une compréhension mathématiquement précise des possibilités et limites fondamentales des ordinateurs et des logiciels. Nous considérons également les implications pratiques de ces limites.

Contenu:

- Introduction aux automates et aux langages formels: automates finis, automates à pile, machines de Turing.
- Introduction à la calculabilité et à la complexité: fonctions récursives, NP-complétude.
- Introduction à la sémantique et à la concurrence: lambda-calcul, algèbres de processus.

Prérequis:

Informatique Théorique I et II

Forme d'enseignement:

Ex cathedra avec exercices

Forme du contrôle:

avec contrôle continu

Bibliographie:

Textbook: Michael Sipser, Introduction to the Theory of Computation, PWS Publishing

Objectives:

This course is an introduction to the "theory of computation".

The goal of this course is to provide a solid and mathematically precise understanding of the fundamental capabilities and limitations of computers and software, as well as their relevance to computer and software engineering practice.

Content:

- Introduction to automata and formal languages: finite automata, push-down automata, Turing machines
- Introduction to computability and complexity: recursive functions, NP-completeness
- Introduction to semantics and concurrency: lambda calculus, process algebra

Required prior knowledge:

Informatique Théorique I et II

Form of teaching:

Ex cathedra with exercices

Form of examination:

with continuous control

Matière examinée / subjects examined		Theoretical computer science III			
Session	PRI	Coefficient / Crédits ECTS	6	Forme de l'examen / Form of examination	Ecrit

Titre	Théorie de l'information
Title	Information theory

Enseignant(s) / Instructor(s)	Chappelier Jean-Cédric: IN	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Informatique (2005-2006, Bachelor semestre 6)	Cours: 2 Exercice: 1		
Mathématiques (2005-2006, Master semestre 2)	Cours: 2 Exercice: 1	3	opt

Objectifs:

Présenter les notions de base de la théorie de l'information et leurs applications dans le codage et la cryptographie.

Contenu:

1. Notions de base: mesures quantitatives de l'incertitude et de l'information propriétés fondamentales de ces mesures
2. Principe de codage d'information
compression de données
codes de Huffman
3. Information en présence d'erreurs
capacité d'un canal
codes correcteurs d'erreurs
codes linéaires par blocs
codes convolutifs
4. Cryptographie
théorèmes fondamentaux
cryptographie à clés secrètes
fonctions à sens unique
cryptographie à clé publique
authentification et signatures numériques

Forme d'enseignement:

Cours on-line avec quelques séances ex cathedra

Bibliographie:

Polycopié du cours
Dominic Welsh: Codes and Cryptography, Oxford Science Publications
Cover & Thomas: Information Theory, Wiley

Objectives:

Introduce basic notions of information theory and their applications in coding and cryptography

Content:

1. Basic notions : quantitative measures of uncertainty and information basic properties of these measures
2. Principles of coding
data compression
Huffman codes
3. Information in the presence of errors
capacity of a medium
error-correcting codes
linear block codes
convolutional codes
4. Cryptography
fundamental theorems
cryptosystems with a secret key
one-way functions
cryptosystems with a public key
authentication and digital signatures

Form of teaching:

On-line course with several ex cathedra sessions

Matière examinée / subjects examined		Théorie de l'information			
Session	ETE	Coefficient / Crédits ECTS	3	Forme de l'examen / Form of examination	Ecrit



ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

SECTION D'INFORMATIQUE

Cycle Master

(Options et spécialisations)

2005 / 2006

Titre	Advanced compiler construction
Title	Advanced compiler construction

Enseignant(s) / Instructor(s)	Schinz Michel: IN	Langue / Language	EN
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Informatique (2005-2006, Master semestre 2)	Cours: 2 Projet: 2	2	opt

Objectifs:

L'étudiant apprendra les techniques utilisées pour l'implémentation de langages de haut niveau et les techniques de compilation utilisées pour obtenir de hautes performances sur les architectures des ordinateurs modernes. Il aura également l'opportunité d'étudier l'une de ces techniques en profondeur et gagnera de l'expérience dans les problèmes d'implémentation au travers d'un projet dans le contexte d'un compilateur actuel.

Contenu:

Techniques d'optimisation :
 Analyse du flot des données, optimisation de programme, génération de code au travers des blocs de base, des procédures et des programmes complets.
 Analyse interprocédurale et intraprocédurale, représentations intermédiaires, allocation de registre et séquençement des instructions.
 Analyse de dépendance et transformations de boucles
 Implémentation de langages de haut niveau
 - Implémentation de fonctions d'ordre supérieur, de coroutines et de processus.
 - Techniques de récupération de la mémoire pour architectures uniprocésseur
 - Machines virtuelles et l'implémentation performante de leur interpréteur.

Un nombre de projets, chacun relié à l'un des sujets ci-dessus sera disponible. Chaque étudiant devra choisir un projet à implémenter, puis écrire un rapport qu'il présentera aux autres étudiants.

Prérequis:

Compiler construction

Forme d'enseignement:

Ex Cathedra. Exercices et Projets en salle et sur l'ordinateur

Forme du contrôle:

Avec contrôle continu

Bibliographie:

Notes polycopiées ou Web

Objectives:

The student will learn about techniques used to implement high level languages, and compilation techniques used to obtain high performance on modern computer architectures. He will also get the opportunity to study one of these techniques in depth and gain experience with implementation issues through a project in the context of an actual compiler.

Content:

Optimization techniques :
 - Data-flow analysis, program optimization, and code generation across basic blocks, procedures, and complete programs.
 - Interprocedural and intraprocedural analysis, intermediate representations, register allocation, and instruction scheduling.
 - Dependence analysis and loop transformations.
 Implementation of high level languages
 - Implementation of higher order functions, coroutines, and processes.
 - Uniprocessor garbage collector techniques.
 - Virtual machines and the efficient implementation of their interpreters.
 A number of projects, each related to one of the above topics, will be available. Each student should choose one project to implement, write a report on, and present to his fellow students.

Required prior knowledge:

Compiler construction

Form of teaching:

Ex Cathedra. Exercises and projects in class and on computer

Form of examination:

With continuous control

Matière examinée / subjects examined		Advanced compiler construction			
Session	ETE	Coefficient / Crédits ECTS	4	Forme de l'examen / Form of examination	Oral

Titre	Advanced computer architecture
Title	Advanced computer architecture

Enseignant(s) / Instructor(s)	Ienne Paolo: IN		Langue / Language	EN
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours	
Informatique (2005-2006, Master semestre 2)	Cours: 2 Projet: 2	6	opt	
Génie électrique et électronique (2005-2006, Master semestre 2)	Cours: 2 Projet: 2	1	opt	
SC - EPFL (IS) (2005-2006, Master semestre 2)	Cours: 2 Projet: 2	6	opt	
SC - EPFL (IS) (2005-2006, Master semestre 4)	Cours: 2 Projet: 2	6	opt	

Objectifs:

Ce cours complète les sujets traités dans les cours « Architecture des ordinateurs I et II ». Les techniques les plus modernes pour l'utilisation du parallélisme au niveau des instructions seront abordées et on discutera de leur relations avec les phases critiques de compilation. Une catégorie de processeurs d'importance croissante - les processeurs pour la conception de systèmes complexes sur un seul circuit intégré - sera aussi analysée ; on discutera à la fois les processeurs commerciaux récents et les dernières directions de recherche

Contenu:

- Augmenter au maximum la performance :
 - o Principes de parallélisme au niveau des instructions
 - o « Register renaming »
 - o Prediction et speculation
 - o Techniques de compilation pour ILP
 - o « Simultaneous multithreading »
 - o « Dynamic binary translation »
 - o Etudes de cas
- Processeurs embarqués VLSI
 - o Particularités par rapport aux processeurs non embarqués
 - o Survol des DSP et des microcontrôleurs pour les Systems-on-Chip
 - o Processeurs configurables et customisation
 - o Problèmes d'implantation VLSI

Prérequis:

Architecture des ordinateurs I et II

Forme d'enseignement:

Ex cathedra

Bibliographie:

J.L. Hennessy et D.A. Patterson, Computer Architecture: A Quantitative Approach, 3rd Edition, 2002.

Objectives:

The course extends and completes the topics of the courses « Computer Architecture I and II ». The most innovative techniques to exploit Instruction-Level Parallelism are surveyed and the relation with the critical phases of compilation discussed. Emerging classes of processors for complex single-chip systems are also analysed by reviewing both recent commercial devices and research directions.

Content:

- Pushing processor performance to its limits:
 - o Principles of Instruction Level Parallelism (ILP)
 - o Register renaming techniques
 - o Prediction and speculation
 - o Compiler techniques for ILP
 - o Simultaneous multithreading
 - o Dynamic binary translation
 - o Case studies
- VLSI embedded processors:
 - o Specificities over stand-alone processors
 - o Overview of DSPs and micro controllers for Systems-on-Chip
 - o Configurable and customisable processors
 - o VLSI design challenges

Required prior knowledge:

Architecture des ordinateurs I et II

Form of teaching:

Ex cathedra

Matière examinée / subjects examined		Advanced computer architecture			
Session	ETE	Coefficient / Crédits ECTS	4	Forme de l'examen / Form of examination	Oral

Titre	Advanced computer graphics
Title	Advanced computer graphics

Enseignant(s) / Instructor(s)	Thalmann Daniel: IN		Langue / Language	EN
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours	
Informatique (2005-2006, Master semestre 2)	Cours: 2 Exercice: 1	3	opt	
SC - EPFL (IS) (2005-2006, Master semestre 2)	Cours: 2 Exercice: 1		opt	
SC - EPFL (IS) (2005-2006, Master semestre 4)	Cours: 2 Exercice: 1		opt	

Objectifs:

Ce cours va expliquer des concepts avancés pour modéliser des objets graphiques complexes, les transformer et leur donner des aspects réalistes. On traitera, en particulier les phénomènes naturels à l'aide de méthodes comme les fractales, les L-systèmes et les systèmes de particules. Dans le domaine du réalisme, on étudiera les problèmes complexes d'ombrage et d'illumination. Enfin, la plus grande partie du cours sera consacrée à l'animation par ordinateur et plus particulièrement aux problèmes complexes de l'animation faciale, de l'animation de foules, de l'animation comportementale, de l'animation de corps déformables incluant les vêtements.

Contenu:

1. MODELISATION GEOMETRIQUE. fractales, L-systèmes, solides
2. RENDU REALISTE. Ombre, réfraction, optimisation du lancer de rayons, radiativité, phénomènes naturels
3. ANIMATION PAR ORDINATEUR. Animation faciale, animation basée sur la physique, animation comportementale, animation de foules, animation de corps déformables, animation de vêtements

Prérequis:

Computer graphics

Forme d'enseignement:

Ex cathedra, films, démos

Forme du contrôle:

avec contrôle continu

Bibliographie:

Notes de cours

Objectives:

This course will explain advanced concepts for modelling of graphical objects, transform them and give them realistic aspects. In particular, we will study natural phenomena using methods like fractals, L-systems, and particle systems. For the rendering, we will emphasize on complex problems of shadowing and lighting. Finally, a large part of the course will be dedicated to computer animation, particularly to problems of facial animation, crowd animation, behavioural animation, animation of deformable bodies, and cloth animation.

Content:

1. GEOMETRIC MODELLING. Fractals, L-systems, solids
2. REALISM. Shadows, refraction, optimization of ray tracing, radiosity, natural phenomena
3. COMPUTER ANIMATION. Facial animation, physics-based animation, behavioral animation, crowd animation, animation of deformable bodies, cloth animation

Required prior knowledge:

Computer graphics

Form of teaching:

Ex cathedra, films, demonstrations

Form of examination:

with continuous control

Matière examinée / subjects examined		Advanced computer graphics			
Session	ETE	Coefficient / Crédits ECTS	4	Forme de l'examen / Form of examination	Ecrit

Titre	Advanced databases
Title	Advanced databases

Enseignant(s) / Instructor(s)	Spaccapietra Stefano: IN		Langue / Language	EN
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours	
Informatique (2005-2006, Master semestre 1)	Cours: 2 Exercice: 2 TP: 2	5	opt	
Informatique (2005-2006, Master semestre 3)	Cours: 2 Exercice: 2 TP: 2	5	opt	
SC - EPFL (IS) (2005-2006, Master semestre 1)	Cours: 2 Exercice: 2 TP: 2	5	opt	
SC - EPFL (IS) (2005-2006, Master semestre 3)	Cours: 2 Exercice: 2 TP: 2	5	opt	

Objectifs:

Ce cours s'adresse aux étudiants qui souhaitent pouvoir s'engager dans des applications avancées utilisant les techniques innovantes des bases de données.
Il forme les étudiants aux concepts et techniques les plus récents des bases de données.

Contenu:

- Etude et analyse critique des systèmes de gestion de bases de données (SGBD) orientés-objets et de leurs langages.
- Etude des SGBD relationnel-objet. Application pratique sur le système Oracle.
- Bases de données dans un environnement distribué: BD réparties, BD fédérées, multi-bases. Application pratique.
- Conception du système d'information dans les systèmes coopératifs: intégration de bases de données.
- Retro-ingénierie de bases de données.
- Modélisation et raisonnement dans les systèmes déductifs.
- Modélisation et fonctionnement des systèmes actifs.
- Systèmes d'information à références spatiales ou temporelles.
- Bases de données sur WEB
- Bases de données multimédias.
- Entrepôts de données. Fouille de données

Prérequis:

Bases de données

Forme d'enseignement:

Ex cathedra; exercices en classe; projets.

Bibliographie:

notes de cours et liste de livres recommandés

Objectives:

This course is intended for those students who aim at being capable of working on new database applications using advanced up to date technology. It covers a wide spectrum of new technologies related to data management.

Content:

- Object-oriented database management systems (DBMSs). Case study. Critical analysis of object-oriented DBMSs and their languages.
- Object-relational DBMSs Case study: Oracle.
- Databases in a distributed environment: distributed databases, federated databases, multidatabases. Case study.
- Database design in cooperative systems: database integration.
- Database reverse engineering.
- Modeling and reasoning in deductive database systems.
- Modeling of active database systems.
- Spatial and temporal information systems.
- Databases on/for the WEB.
- Multimedia Databases.
- Data Warehousing, Data Mining.

Required prior knowledge:

Bases de données

Form of teaching:

Ex cathedra; exercises in class; projects

Matière examinée / subjects examined		Advanced databases			
Session	PRI	Coefficient / Crédits ECTS	6	Forme de l'examen / Form of examination	Ctrl continu

Titre	Advanced digital design
Title	Advanced digital design

Enseignant(s) / Instructor(s)	Sanchez Eduardo: IN		Langue / Language	EN
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours	
Informatique (2005-2006, Master semestre 2)	Cours: 4 Exercice: 2	6	opt	
SC - EPFL (IS) (2005-2006, Master semestre 2)	Cours: 4 Exercice: 2	6	opt	
SC - EPFL (IS) (2005-2006, Master semestre 4)	Cours: 4 Exercice: 2	6	opt	

Objectifs:

Connaissance et utilisation des méthodes et des outils de conception des systèmes numériques complexes.

Contenu:

Synthèse de systèmes logiques multiniveaux: méthodologie et utilisation d'outils CAO.

Circuits programmables à grande complexité: étude et utilisation de différentes familles de circuits FPGA.

Langages de description et de simulation de matériel: VHDL.

Synthèse automatique: génération des schémas logiques à partir des descriptions fonctionnelles en VHDL.

Synthèse architecturale: co-design. Conception globale d'un système, avec une partie logicielle (programme exécuté par un processeur) et une partie matérielle (circuit programmable ou circuit intégré spécifique).

Systèmes reconfigurables.

Exemples: réalisation d'un contrôleur de mémoire cache, réalisation d'un processeur superscalaire, etc.

Forme d'enseignement:

Ex cathedra; exercices en salle de stations

Bibliographie:

Notes polycopiées

Objectives:

Knowledge and use of methods and tools for the development of complex digital systems

Content:

Synthesis of multi-level logic systems: methodology and use of CAD tools.

High-complexity programmable circuits: study and use of different families of FPGA circuits.

Hardware description and simulation languages:VHDL.

Automatic synthesis: generation of logic schematics from functional description in VHDL.

Architectural synthesis: co-design. Complete development of a system, with a software part (program executed by a processor) and a hardware part (programmable or custom integrated circuit).

Reconfigurable systems.

Examples: realization of a cache memory controller, realization of a superscalar processor, etc.

Form of teaching:

Ex cathedra; exercises in computer room

Matière examinée / subjects examined		Advanced digital design			
Session	ETE	Coefficient / Crédits ECTS	6	Forme de l'examen / Form of examination	Ctrl continu

Titre	Color reproduction
Title	Color reproduction

Enseignant(s) / Instructor(s)	Hersch Roger: IN	Langue / Language	EN
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Informatique (2005-2006, Master semestre 2)	Cours: 2 Projet: 2	3	opt
SC - EPFL (IS) (2005-2006, Master semestre 2)	Cours: 2 Projet: 2	2	opt
SC - EPFL (IS) (2005-2006, Master semestre 4)	Cours: 2 Projet: 2	2	opt

Objectifs:

Ce cours donne une introduction à la colorimétrie et présente les éléments permettant de modéliser numériseurs, dispositifs d'affichage et imprimantes couleur. La reproduction d'image en demi-tons ainsi que les procédés de modélisation et de calibration d'imprimantes sont traités de manière approfondie. Les notions acquises sont utiles pour comprendre certaines techniques de protection contre la contrefaçon.

Contenu:

Fondements de la colorimétrie

Sensibilité spectrale des récepteurs rétinaux, égalisation colorimétrique, les systèmes CIE-XYZ, xyY, CIE-LAB, RGB, YIQ, CMYK, systèmes additifs et soustractifs, mesures spectrales.

Interaction entre lumière et papier imprimé

Loi de Beer, correction de Saunderson (reflexions multiples).

Périphériques couleur

Modélisation des numériseurs, écrans, et imprimantes, impression noir/blanc et couleur, séparation couleur, calibration d'une chaîne de reproduction couleur, mise en correspondance de volumes couleur, modèles prédictifs de Neugebauer, Yule-Nielson et Clapper-Yule.

Génération d'images en demi-tons (halftoning)

Procédés de génération d'images tramées: points groupés, super-trames, points dispersés, diffusion d'erreurs, phénomènes de moirés, trames couleur, trames à microstructures.

Les laboratoires s'effectueront en MatLab et permettront d'exercer les notions présentées au cours. Un mini-projet permettra d'approfondir les notions acquises.

Forme d'enseignement:

Cours, laboratoires sur ordinateur (Matlab), mini-projet

Remarque:

Les cours "Color Imaging" et "Color Reproduction" sont indépendants, mais complémentaires. Il est recommandé de les prendre dans cet ordre.

Bibliographie:

Color reproduction, notes de cours et de laboratoires

Objectives:

The course introduces the fundamentals of colorimetry, as well as models for scanners, displays and printers. The main focus is on halftoning and color reproduction (color separation, gamut mapping, color prediction for printing devices). The introduced concepts are useful for the understanding of anti-counterfeiting methods (protective features for banknotes, checks, etc).

Content:

Color theory

Spectral sensibility of the eye, colorimetric equalization, the CIE-XYZ, xyY, CIE-LAB, RGB, YIQ, CMYK systems, additive and subtractive systems, spectral measurements.

Interaction between light and printed paper

Beer's law, the Saunderson correction (multiple reflections).

Color devices

Modellization of scanners, displays and printers, black-white and color printing, density measurements, dot-gain, color separation, device calibration (scanner, display, printer), gamut mapping, Color prediction models (Neugebauer, Yule-Nielson, Clapper-Yule).

Halftoning algorithms

Clustered-dot dithering, dispersed-dot dithering, supercells, error diffusion, moiré phenomena between color layers, color halftoning, microstructure imaging,

The course is coupled with laboratories in MatLab which enable exercising the concepts presented during the lectures. A small project enables each student to gain concrete experience with some of the course's topics.

Form of teaching:

Course, computer laboratories (Matlab), short-project

Note:

The courses "Color Imaging" and "Color Reproduction" are independent, but complementary. They may be taken in that order.

Matière examinée / subjects examined		Color reproduction			
Session	ETE	Coefficient / Crédits ECTS	4	Forme de l'examen / Form of examination	Oral

Titre	Complex circuits
Title	Complex circuits

Enseignant(s) / Instructor(s)	Beuchat René: IN, Piguet Christian: IN		Langue / Language	EN
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours	
Informatique (2005-2006, Master semestre 1)	Cours: 2 Projet: 2	6	opt	
Informatique (2005-2006, Master semestre 3)	Cours: 2 Projet: 2	6	opt	
Génie électrique et électronique (2005-2006, Master semestre 1)	Cours: 2 Projet: 2	1	opt	
SC - EPFL (IS) (2005-2006, Master semestre 1)	Cours: 2 Projet: 2	6	opt	
SC - EPFL (IS) (2005-2006, Master semestre 3)	Cours: 2 Projet: 2	6	opt	

Objectifs:

La technologie VLSI a permis le développement des processeurs et mémoires, et doit encore s'améliorer d'un facteur 1000 dans les 15 prochaines années. Le but du cours est de comprendre l'influence de la technologie et surtout des contraintes de consommation sur l'architecture des systèmes sur chip comportant des microcontrôleurs, microprocesseurs, mémoires, mémoires cache, DSP et machines parallèles. Dans tout système sur chip, les mémoires et les bus sont de toute première importance pour les performances tant en vitesse qu'en consommation.

Le cours suppose une bonne connaissance des architectures de processeurs et périphériques. Il prépare pour des projets de systèmes sur chip et systèmes sur cartes avec développement de circuits intégrés spécifiques.

Contenu:

- Evolution des technologies VLSI
- Prédications de la Roadmap SIA 2001-2016
- Futures technologies et nouvelles techniques de circuits
- Circuits asynchrone et adiabatique
- Microcontrôleurs basse consommation
- Microprocesseurs basse consommation
- Mémoires et caches basse consommation
- DSP et machines parallèles basse consommation
- Mémoires dynamiques DRAM de haute complexité
- Circuits interfaces pour bus parallèle et série
- Interfaces processeur-mémoire, asynchrone et synchr

Prérequis:

Systèmes Microprocesseurs, Conception de Systèmes numériques

Forme d'enseignement:

Ex cathedra

Objectives:

VLSI technology allows the development of processors and memories. Significant improvements, by a factor 1000 or more, are still expected over the next 15 years. The objective of the course is to understand the influence of technology and mainly power consumption constraints on the architecture of microcontrollers, microprocessors, memories, cache memories, DSP and parallel machines. In any system on chip, memories and buses are very important for achieving speed and power consumption performances.

The course supposes a good knowledge of processor and I/O architectures. Students will be prepared to develop systems on chip and on boards with development of specific integrated circuits.

Content:

- Evolution of VLSI technologies
- SIA Roadmap predictions (2001-2016)
- Future technologies and new circuit techniques
- Asynchronous and adiabatic circuits
- Low-power microcontrollers
- Low-power microprocessors
- Low-power memories and cache memories
- Low-power DSP and parallel machines
- Complex dynamic DRAM memories
- Circuit interfaces or parallel and serial buses
- Asynchronous - synchronous processor-memory interfaces

Required prior knowledge:

Systèmes Microprocesseurs, Conception de Systèmes numériques

Form of teaching:

Ex cathedra

Matière examinée / subjects examined		Complex circuits			
Session	PRI	Coefficient / Crédits ECTS	4	Forme de l'examen / Form of examination	Oral

Titre	Computational genomics
Title	Computational genomics

Enseignant(s) / Instructor(s)	Galisson Frédérique: IN		Langue / Language	EN
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours	
Informatique (2005-2006, Master semestre 1)	Cours: 4 Exercice: 2	1	opt	
Informatique (2005-2006, Master semestre 3)	Cours: 4 Exercice: 2	1	opt	
SC - EPFL (IS) (2005-2006, Master semestre 1)	Cours: 4 Exercice: 2	4	opt	
SC - EPFL (IS) (2005-2006, Master semestre 3)	Cours: 4 Exercice: 2	4	opt	

Objectifs:

- Rappeler les concepts fondamentaux en biologie, en particulier en génétique moléculaire et évolution ; - Expliquer les changements méthodologiques en biologie moléculaire avec les projets de séquençage et mesure de l'expression des gènes à grande échelle, et donner une intro. générale à la bioinformatique, centrée sur les aspects informatiques de la génomique ; - Etudier les algorithmes et modèles mathématiques importants, utilisés pour inférer des connaissances biologiques à partir des données de séquence et d'expression, notamment les outils d'intérêt général.

Contenu:

1. Présentation historique de la biologie moderne ; 2. Concepts de base en biologie cellulaire, biochimie et génétique ; 3. Biologie moléculaire : ADN, ARN, gènes, l'information génétique et son expression ; 4. Séquençage de génomes, génomique, et les récents changements méthodologiques de la recherche en biologie ; 5. Comparaison de séquences biologiques : algorithmes d'alignement (Programmation Dynamique) ; modèles d'évolution moléculaire et systèmes de scores dérivés ; 6. Comparaisons à grande échelle ; 7. Alignements multiples et motifs (1) : algorithmes pour l'alignement multiple global ; 8. Alignements multiples et motifs (2) : modélisation de l'information contenue dans un alignement multiple : expressions régulières, profils statistiques, Modèles de Markov Caches (HMM) et algorithmes associés ; 9. Alignements multiples et motifs (3) : Inférence de motifs à partir d'ensemble de séquences non alignées : approches déterministes ou probabilistes ; 10. L'algorithme EM (« Esperance-Maximisation ») : présentation générale et application à l'inférence de motifs et à l'apprentissage non supervisé des paramètres de HMM ; 11. Prédiction de gènes : modélisation des biais statistiques des régions codantes, modèles probabilistes de la structure des gènes, et algorithmes associés ; 12. Inférence phylogénétique ; 13. Transcriptome et mesure de l'expression des gènes à grande échelle : technologie des puces à ADN et exemples de questions qu'elle permet d'aborder ; 14. Analyse des données d'hybridation de puces à ADN : modèles de l'expression des gènes, algorithmes de classification, retour à l'algorithme EM.

Forme d'enseignement:

Exercices sur papier et à l'ordinateur

Forme du contrôle:

avec contrôle continu

Bibliographie:

- Purves et al., "Life: the science of biology", vol.1, Sinauer Associates/W.H. Freeman
- David W. Mount, "Bioinformatics -- Sequence and Genome Analysis", Cold Spring Harbor Laboratory Press
- Richard Durbin et al., "Biological sequence analysis - Probabilistic models of proteins and nucleic acids", Cambridge University Press (1998).
- Pierre Baldi and Søren Brunak, "Bioinformatics: the Machine Learning Approach" -- 2nd edition, MIT Press, (2000).

Objectives:

- Recall fundamental concepts in biology, in particular about molecular genetics and evolution ; present molecular sequence and expression data ; - Explain the methodological changes which occurred in molecular biology with the advent of large scale sequencing and gene expression monitoring projects, and provide a general introduction to computational molecular biology with a focus on computational genomics ; - Study important algorithms and models used for inferring biological knowledge from sequence and expression data, with an emphasis on general purpose tools.

Content:

1. Historical presentation of modern biology ;
2. Basic concepts in cellular biology, biochemistry and genetics ;
3. Molecular biology : DNA, RNA, genes, genetic information, its expression and evolution ;
4. Genome sequencing, genomics, and the subsequent recent changes in biological research methods.
5. Biological sequence comparison : pairwise sequence alignments (Dynamic Programming) ; models of molecular evolution and derived scoring systems ; 6. Large scale sequence comparisons ;
7. Multiple alignment and sequence motifs (1) : algorithms for global multiple alignment ;
8. Multiple alignment and sequence motifs (2) : modeling the information contained in multiple alignments : regular expressions, statistical profiles, Hidden Markov Models (HMM), and associated algorithms ;
9. Multiple alignment and sequence motifs (3) : pattern inference from sets of unaligned sequences : deterministic or probabilistic approaches ;
10. The Expectation-Maximisation (EM) algorithm : general presentation and specific application to HMM parameter estimation and sequence motif inference ;
11. Gene prediction : modeling coding region biases, probabilistic models of gene structure, and associated algorithms ; 12. Phylogenetic inference ;
13. Transcriptome and measurement of gene expression levels : technology of DNA microarrays and examples of questions addressed through this technology ;
14. Microarray data analysis : models of gene expression, clustering algorithms, back to EM, with model-based clustering .

Form of teaching:

Exercices on paper and on computer

Form of examination:

with continuous control

Matière examinée / subjects examined		Computational genomics			
Session	PRI	Coefficient / Crédits ECTS	6	Forme de l'examen / Form of examination	Ecrit

Titre	Computational processing of textual data
Title	Computational processing of textual data

Enseignant(s) / Instructor(s)	Chappelier Jean-Cédric: IN, Rajman Martin: IN	Langue / Language	EN
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Informatique (2005-2006, Master semestre 2)	Cours: 4 Exercice: 2	5	opt
SC - EPFL (IS) (2005-2006, Master semestre 2)	Cours: 4 Exercice: 2	5	opt
SC - EPFL (IS) (2005-2006, Master semestre 4)	Cours: 4 Exercice: 2	5	opt

Objectifs:

Manipuler de façon efficace les données textuelles est devenu une nécessité pour les systèmes d'information modernes. Dans des applications comme les moteurs de recherche sur le Web, les systèmes d'extraction d'information (Text Mining) ou plus simplement les systèmes avancés de traitement de documents (correction, résumé, traduction, ...), l'utilisation de techniques sensibles au contenu linguistique constitue aujourd'hui un avantage concurrentiel certain.

L'objectif de ce cours est de présenter les principaux modèles, formalismes et algorithmes permettant l'intégration de techniques d'informatique linguistique dans les applications d'informatique documentaire. Les concepts introduits en cours seront mis en pratique lors de TP.

Contenu:

Divers modèles et algorithmes génériques pour le traitement de données textuelles seront présentés : (1) niveau morpho-lexical : lexiques informatiques, correction orthographique, ...; (2) niveau syntaxique : grammaires régulières, non-contextuelles, stochastiques ; algorithmes d'analyse syntaxique ; ...; (3) niveau sémantique : modèles et formalismes pour la représentation du sens, (4) niveau pragmatique : modèles et formalismes pour la gestion de dialogues, interprétation contextuelle, actes de langage. Plusieurs domaines pratiques seront abordés : Ingénierie linguistique, Recherche Documentaire, Text-Mining (extraction automatique de connaissances), Analyse des données textuelles (classification automatique de documents, visualisation de bases de données textuelles). Certains des cours magistraux pourront être donnés en anglais en fonction de l'auditoire.

Forme d'enseignement:

Ex cathedra; travaux pratiques sur ordinateur

Remarque:

Pour plus d'information sur le cours, voir site Web

Bibliographie:

Notes de cours

Objectives:

Efficient handling of textual data is an important requirement for modern information systems. In applications such as search engines on the Web, Text Mining systems (information extraction) or even advanced document processing systems (correction, summary, translation...), the use of techniques sensitive to the linguistic content represents nowadays a clear competitive advantage.

The objective of this course is to present the main models, formalisms and algorithms necessary for the development of applications in the field of documentary information processing. The concepts introduced during the lectures will be applied during practical sessions.

Content:

Several models and algorithms for automated textual data processing will be described: (1) morpho-lexical level: electronic lexica, spelling checkers, ...; (2) syntactic level: regular, context-free, stochastic grammars, parsing algorithms, ...; (3) semantic level: models and formalisms for the representation of meaning, ...; (4) pragmatic level: models and formalisms for dialogue management, contextual interpretation, speech acts.

Several application domains will be presented: Linguistic engineering, Information Retrieval, Text mining (automated knowledge extraction), Textual Data Analysis (automated document classification, visualization of textual data).

Form of teaching:

Ex cathedra; practical work on computer

Note:

For further details, see Web site:

URLs	1) http://icwww.epfl.ch/~chappeli/tidt				
Matière examinée / subjects examined	Computational processing of textual data				
Session	ETE	Coefficient / Crédits ECTS	6	Forme de l'examen / Form of examination	Ecrit

Titre	Computer-aided verification
Title	Computer-aided verification

Enseignant(s) / Instructor(s)	Henzinger Thomas: IN		Langue / Language	EN
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours	
Informatique (2005-2006, Master semestre 2)	Cours: 4	2 6		
SC - EPFL (IS) (2005-2006, Master semestre 2)	Cours: 4	6	opt	
SC - EPFL (IS) (2005-2006, Master semestre 4)	Cours: 4	6	opt	

Objectifs:

Acquérir les bases théoriques et pratiques de la vérification formelle software et hardware, ceci en particulier à travers l'étude d'une technique dite de « vérification de modèle ».

Contenu:

La vérification de modèle concerne l'utilisation d'algorithmes assurant la sûreté et la performance des systèmes software et hardware. Les systèmes numériques occupants une place de plus en plus prépondérante dans notre quotidien, la fiabilité de tels systèmes est devenu une question d'importance majeure. Lorsque la complexité de ces systèmes grandit, leur fiabilité ne peut malheureusement plus être assurée par les approches traditionnelles de test et de simulation.

Le cours abordera les sujets suivants :

1. Modélisation de systèmes : concurrence, temps réel, sécurité / vivacité.
2. Algorithmes de vérification : logique temporelle, automates, jeux.
3. Problèmes de dimensionnement : méthodes symboliques, modularité, abstraction.
4. Sujets avancés : systèmes hybrides, systèmes stochastiques.

Objectives:

The participants will become familiar with both the theory and practice of formal software and hardware verification, in particular with the technique called model checking.

Content:

Model checking concerns the use of algorithms for the safety and performance assurance of software and hardware systems. As our daily lives depend increasingly on digital systems, the reliability of these systems becomes a concern of overwhelming importance, and as the complexity of the systems grows, their reliability can no longer be sufficiently controlled by the traditional approaches of testing and simulation.

The course will cover the following topics.

1. System modeling: concurrency, real time, safety vs. liveness.
2. Verification algorithms: temporal logic, automata, games.
3. Scalability issues: symbolic methods, modularity, abstraction.
4. Advanced topics: hybrid systems, stochastic systems.

Matière examinée / subjects examined		Computer-aided verification			
Session	ETE	Coefficient / Crédits ECTS	6	Forme de l'examen / Form of examination	Ctrl continu

Titre	Computer-supported cooperative work
Title	Computer-supported cooperative work

Enseignant(s) / Instructor(s)	Dillenbourg Pierre: IN		Langue / Language	EN
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours	
Informatique (2005-2006, Master semestre 1)	Cours: 2 Exercice: 2		opt	
Informatique (2005-2006, Master semestre 3)	Cours: 2 Exercice: 2		opt	
Programme Management de la technologie et entrepreneuriat (2005-2006, Master semestre 1)	Cours: 2 Exercice: 2		opt	
Programme Management de la technologie et entrepreneuriat (2005-2006, Mineur semestre 1)	Cours: 2 Exercice: 2		opt	

Objectifs:

L'objectif de ce cours est que les étudiants soient capables de:

- Spécifier un roomware selon le processus collaboratif à faciliter. Cet objectif nécessite l'analyse des processus de collaboration et une compréhension de la relation entre les caractéristiques du software (architecture et fonctionnalités) et du hardware ainsi que les processus de collaboration
- Analyser l'impact d'un collecticiel sur les processus collaboratifs. Cet objectif nécessite la construction d'une expérience pour mesurer les modifications au sein du travail en groupe et analyser les interactions au moyen de méthodes quantitative ou qualitatives.

Contenu:

- Impact des collecticiels sur la collaboration
- Questions d'ergonomie dans les collecticiels
- Roomware & ubiquitous computing
- Théories de la cognition distribuée
- Méthodologie de recherche expérimentale
- Techniques d'analyse de données

Nous aborderons différents types de collecticiels tels que les 'workflows', le 'roomware' les éditeurs partagés, les environnements virtuel de collaboration et d'apprentissage collaboratif et les collecticiels reposant sur les technologies mobiles.

Prérequis:

Human-Computer Interaction (Pu) - recommandé

Forme d'enseignement:

Research project + lectures

Forme du contrôle:

avec contrôle continu

Bibliographie:

Dix, A., Finlay, J., Abowd, G., and Beale, R. (1998) Chapter 13: Groupware. In Human Computer Interaction, 2nd Edition. 463-508, Prentice Hall.

Objectives:

The goal of the CSCW course is that students become able:

- To specify a groupware system that supports specific teamwork processes. This requires analyzing the process to be supported and understanding the relationship between groupware features (architecture and functionalities), hardware features and cooperative processes.
- To analyze the impact of a groupware system on the cooperative processes. This requires constructing an experiment for measuring changes within group processes and analyzing team interactions with both qualitative and quantitative data analysis techniques.

Content:

- Groupware typology and functionalities
- Architectures and models in CSCW
- CSCW issues in human-computer interaction
- Group processes and organizational factors Distributed cognition theories
- Methodology of experimental research
- Data analysis techniques.

Many groupware systems will be considered such as workflows, roomware, shared editors, virtual collaborative environments, media spaces, computer-supported collaborative learning environments and CSCW environments based on mobile devices.

Required prior knowledge:

Human-Computer Interaction (Pu) - recommandé

Form of teaching:

Research project + lectures

Form of examination:

with continuous control

Matière examinée / subjects examined		Computer-supported cooperative work			
Session	PRI	Coefficient / Crédits ECTS	6	Forme de l'examen / Form of examination	Oral

Titre	Computer vision
Title	Computer vision

Enseignant(s) / Instructor(s)	Fua Pascal: IN	Langue / Language	EN
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Informatique (2005-2006, Master semestre 2)	Cours: 2 Exercice: 1	3	opt
SC - EPFL (IS) (2005-2006, Master semestre 2)	Cours: 2 Exercice: 1	2	opt
SC - EPFL (IS) (2005-2006, Master semestre 4)	Cours: 2 Exercice: 1	2	opt

Objectifs:

L'étudiant pourra identifier le type de problèmes posés par la vision par ordinateur et saura mettre en oeuvre des méthodes adéquates de traitement d'image.

La vision par ordinateur est la branche de l'informatique qui tente de modéliser le monde réel ou de reconnaître des objets à partir d'images digitales. Ces images peuvent être acquises par des caméras vidéos, infrarouges, des radars ou des senseurs spécialisés tels ceux utilisés par les médecins.

Nous nous concentrerons sur le traitement d'images noir et blanc ou couleur obtenues par des caméras vidéo classiques et nous introduirons les techniques de base.

Contenu:

Introduction

- Historique de la vision par ordinateur.
- Vision humaine et Vision par Ordinateur
- Formation des images

Analyse d'images en deux dimensions

- Espace des échelles
- Détection de contours
- Suivi d'objets
- Segmentation niveaux de gris, couleur et texture

La troisième dimension

- Ombrage
- Stéréographie
- Silhouettes
- Mouvement

Forme d'enseignement:

Ex cathedra, films et exercices sur ordinateur

Forme du contrôle:

Avec contrôle continu

Bibliographie:

- V. S. Nalwa, A Guided Tour of Computer Vision, Addison-Wesley, 1993.
D. A. Forsyth, J. Ponce, Computer Vision: A Modern Approach, Prentice Hall, 2002

Objectives:

The student will be introduced to the basic techniques of the field of Computer Vision. He will learn to apply Image Processing techniques where appropriate.

Computer Vision is the branch of Computer Science whose goal is to model the real world or to recognize objects from digital images. These images can be acquired using video or infrared cameras, radars or specialized sensors such as those used by doctors.

We will concentrate on the black and white and color images acquired using standard video cameras. We will introduce the basic processing techniques.

Content:

Introduction

- History of Computer Vision
- Human vs Machine Vision
- Image formation

2-D Image Analysis

- Scale-space
- Delineation
- Tracking
- Gray-level, color and texture segmentation

3-D Image Processing

- Shading
- Stereo
- Silhouettes
- Motion

Form of teaching:

Ex cathedra, films and exercises on computer

Form of examination:

With continuous control

Matière examinée / subjects examined		Computer vision			
Session	ETE	Coefficient / Crédits ECTS	4	Forme de l'examen / Form of examination	Ecrit

Titre	Cryptography and security
Title	Cryptography and security

Enseignant(s) / Instructor(s)	Oechslin Philippe: SC, Vaudenay Serge: SC		Langue / Language	EN
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours	
Informatique (2005-2006, Master semestre 1)	Cours: 4 Projet: 2	5	opt	
Informatique (2005-2006, Master semestre 3)	Cours: 4 Projet: 2	5	opt	
SC - EPFL (IS) (2005-2006, Master semestre 1)	Cours: 4 Projet: 2	3 5 7	obl	
SC - EPFL (IS) (2005-2006, Master semestre 3)	Cours: 4 Projet: 2	3 5 7	obl	

Objectifs:

Comprendre les menaces contre les réseaux informatiques, savoir comment les protéger par des mesures techniques ou organisationnelles. Introduire les bases de la cryptographie : comment l'implémenter, comment l'utiliser.

Contenu:

1. Cryptographie conventionnelle :
 - chiffrement par blocs, modes opératoires, chiffrement par flots, fonctions de hachage, codes d'authentification de message
 - attaques par force brute, paradoxe des anniversaires
 - application au contrôle d'accès
2. Cryptographie à clef publique :
 - RSA: cryptosystème à clef publique, exemple de problèmes de sécurité, signature numérique
 - protocole de Diffie-Hellman, chiffrement et signature de ElGamal
3. Aspects techniques :
 - attaque communes : virus, chevaux de Troie, déni de service, crackage
 - mesures de protection : filtres, pare-feus, proxys, anti-virus, détection d'intrusion
 - protocoles : IPSec, HTTPS, SSL/TLS, PGP, S/MIME, SSH, PPTP
4. Aspects organisationnels :
 - analyse de risque et politiques de sécurité
 - audit de sécurité
5. Aspects humains et de régulation :
 - aspects légaux sur la sphère privée et la protection de la propriété intellectuelle
 - éthique, sensibilisation, dissuasion

Forme d'enseignement:

Ex cathedra

Forme du contrôle:

Avec contrôle continu

Bibliographie:

Communication Security : an introduction to cryptography. Serge Vaudenay

Objectives:

To understand the threats to which computer networks are exposed, to know how to protect a network using appropriate technical and organisational measures. To introduce basic cryptography: how to implement it, how to use it.

Content:

1. Conventional cryptography:
 - block ciphers, modes of operation, stream ciphers, hash functions, message authentication codes
 - brute force attacks, birthday paradox
 - applications to access control
2. Public key cryptography:
 - RSA: public key cryptosystem, example of security faults, digital signature
 - Diffie-Hellman protocol, ElGamal encryption and signature
3. Technical aspects:
 - common attacks: virus, Trojan horse, denial of service, cracking
 - protective measures: filters, firewalls, proxys, anti-virus, intrusion detection
 - protocols: IPSec, HTTPS, SSL/TLS, PGP, S/MIME, SSH, PPTP
4. Organisational aspects:
 - risk analysis and security policies
 - security inspection and audit
5. Regulation and human aspects:
 - legal aspects related to privacy, intellectual property protection
 - ethics, awareness, dissuasion

Form of teaching:

Ex cathedra

Form of examination:

With continuous control

Matière examinée / subjects examined		Cryptography and security			
Session	PRI	Coefficient / Crédits ECTS	7	Forme de l'examen / Form of examination	Ecrit

Titre	Design technologies for intergrated systems
Title	Design technologies for intergrated systems

Enseignant(s) / Instructor(s)	De Micheli Giovanni: IN		Langue / Language	EN
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours	
Informatique (2005-2006, Master semestre 1)	Cours: 3 Projet: 2	6	opt	
Informatique (2005-2006, Master semestre 3)	Cours: 3 Projet: 2	6	opt	
SC - EPFL (IS) (2005-2006, Master semestre 1)	Cours: 3 Projet: 2	6	opt	
SC - EPFL (IS) (2005-2006, Master semestre 3)	Cours: 3 Projet: 2	6	opt	

Objectifs:

Les étudiants étudieront les techniques utilisées pour la conception des circuits et systèmes intégrés en partant d'une description formelle à l'aide de langages de conception hardware et l'optimisation des circuits digitaux en terme de porte logique.

Contenu:

La synthèse hardware est effectuée grâce à la transformation d'un langage spécialisé de description hardware en une description de circuits, qui est affinée et optimisée par itérations successives. Ce cours présentera les principales spécificités de la synthèse hardware et les différentes techniques d'optimisation des représentations logiques. Ce cours donne une vision nouvelle et actuelle de la conception de circuits digitaux.

Les travaux pratiques montreront aux étudiants l'utilisation des outils de conception principaux.

Programme

- 1) Langages de modélisation et de spécification
- 2) Synthèse haut niveau et méthodes d'optimisation (planification, liaison, chemin de données et contrôle)
- 3) Représentation et optimisation de fonctions logique combinatoires (problème d'encodage, diagrammes de décision binaire)
- 4) Représentation et optimisation de réseau à couche multiple (méthodes algébriques et booléennes, calcul des ensembles « don't care », vérification et optimisation des temps de propagation)
- 5) Modélisation et optimisation de fonctions séquentielles et de réseaux (retiming)
- 6) Librairies partiellement personnalisées et liaison de librairies.

Forme d'enseignement:

Ex cathedra

Bibliographie:

Synthesis and Optimization of Digital Circuits by Pr. G. De Micheli

Objectives:

Students will learn the techniques used for designing integrated circuits and systems starting from design languages and formalism to the synthesis and optimization of digital circuits in terms of logic gates.

Content:

Hardware compilation is the process of transforming specialized hardware description languages into circuit descriptions, which are iteratively refined, detailed and optimized. The course will present the most outstanding features of hardware compilation, as well as the techniques for optimizing logic representations and networks. The course gives a novel, uptodate view of digital circuit design. Practical sessions will teach students the use of current design tools.

Syllabus

- 1) Modeling languages and specification formalisms;
- 2) High-level synthesis and optimization methods (scheduling, binding, data-path and control synthesis);
- 3) Representation and optimization of combinational logic functions (encoding problems, binary decision diagrams);
- 4) Representation and optimization of multiple-level networks (algebraic and Boolean methods, "don't care" set computation, timing verification and optimization);
- 5) Modeling and optimization of sequential functions and networks (retiming);
- 6) Semicustom libraries and library binding.

Form of teaching:

Ex cathedra

Matière examinée / subjects examined		Design technologies for intergrated systems		
Session	PRI	Coefficient / Crédits ECTS	6	Forme de l'examen / Form of examination
				Ctrl continu

Titre	Distributed algorithms
Title	Distributed algorithms

Enseignant(s) / Instructor(s)	Schiper André: SC		Langue / Language	EN
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours	
Informatique (2005-2006, Master semestre 1)	Cours: 2 Exercice: 1	2 5	opt	
Informatique (2005-2006, Master semestre 3)	Cours: 2 Exercice: 1	2 5	opt	
SC - EPFL (IS) (2005-2006, Master semestre 1)	Cours: 2 Exercice: 1	5	obl	
SC - EPFL (IS) (2005-2006, Master semestre 3)	Cours: 2 Exercice: 1	5	obl	

Objectifs:

Les algorithmes répartis constituent l'algorithmique fondamentale de nombreuses applications et systèmes de communication. On peut citer par exemple les applications de réservation, la finance, le contrôle de trafic aérien, ainsi que la gestion des systèmes de communication.

L'objectif de ce cours est de présenter aux étudiants les fondements des algorithmes répartis et de leur apprendre à aborder de manière rigoureuse les problèmes de distribution et leurs solutions.

Contenu:

1. Modèles

Mémoire partagée vs. échange de messages, système synchrone, système asynchrone, système partiellement synchrone, autres aspects.

2. Consensus

Consensus dans un modèle synchrone, impossibilité du consensus dans un système asynchrone, consensus dans un modèle avec recovery, algorithmes probabilistes, défaillances byzantines.

3. Validation atomique et lien avec consensus

4. Diffusion avec diverses garanties

Diffusion fiable probabiliste, communication épidémique, classes d'algorithmes de diffusion atomique, algorithmes de diffusion générique.

5. Problème du «Group membership»

Résultat d'impossibilité, modèle de partition primaire, modèle partitionné.

6. Algorithmes auto-stabilisants

Principe de l'auto-stabilisation, différence entre auto-stabilisation et autres techniques de tolérance aux défaillances, exemples d'algorithmes auto-stabilisants.

7. Mémoire partagée

Equivalence entre modèle à échange de messages et modèle à mémoire partagée, objets atomiques, critères de cohérence.

Forme d'enseignement:

Ex cathedra

Bibliographie:

Sera annoncé en classe

Objectives:

Distributed algorithms are at the heart of many applications and communication systems. These include banking, reservation and air traffic control applications, as well as network management systems. The aim of this course is to expose the students to the fundamentals of distributed algorithms and teach them how to approach and reason in a rigorous manner about problems of distribution and their solutions.

Content:

1. Models

Shared memory vs. message exchange, synchronous system, asynchronous system, partially synchronous system, other aspects.

2. Consensus

Consensus in a synchronous system, impossibility of solving consensus in an asynchronous system, consensus in a model with recovery, probabilistic algorithms, byzantine failures.

3. Atomic commitment and its link with consensus

4. Broadcast with various guarantees

Probabilistic broadcast, epidemic communication, classes of atomic broadcast algorithms, algorithms for generic broadcast.

5. Group membership problem

Impossibility result, primary partition model, partitionable model.

6. Self-stabilizing algorithms

Principle for self-stabilization, difference between self-stabilization and other fault-tolerant techniques, examples of self-stabilization algorithms.

7. Shared memory

Equivalence between message exchange model and shared memory model, atomic objects, consistency criteria.

Form of teaching:

Ex cathedra

Matière examinée / subjects examined		Distributed algorithms			
Session	PRI	Coefficient / Crédits ECTS	4	Forme de l'examen / Form of examination	Ecrit

Titre	Distributed information systems
Title	Distributed information systems

Enseignant(s) / Instructor(s)	Aberer Karl: SC		Langue / Language	EN
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours	
Informatique (2005-2006, Master semestre 1)	Cours: 2 Exercice: 1	1 5	opt	
Informatique (2005-2006, Master semestre 3)	Cours: 2 Exercice: 1	1 5	opt	
SC - EPFL (IS) (2005-2006, Master semestre 1)	Cours: 2 Exercice: 1	3 4 5	obl	
SC - EPFL (IS) (2005-2006, Master semestre 3)	Cours: 2 Exercice: 1	3 4 5	obl	

Objectifs:

La conférence donne une vue d'ensemble des problèmes principaux dans la gestion Enchaînement-basée et mobile de l'information. Elle présente en détail un choix des approches caractéristiques, de la pratique et de la recherche, et crée ainsi la prise de conscience pour les défis spécifiques dans la gestion distribuée de l'information et les solutions typiques. Les étudiants pourront identifier les différentes classes de problème dans la gestion distribuée de l'information (par exemple gestion des données mobile) et les techniques correspondantes pour les résoudre (par exemple des structures d'indexation), pour comprendre de diverses méthodes standard dans la gestion distribuée de l'information (par exemple recherche documentaire de l'espace de vecteur) et pour s'appliquer ces méthodes aux problèmes pratiques (simples). Nous procédons aux niveaux croissants de l'abstraction. Nous commençons à partir des aspects physiques des données distribuées et mobiles de gestion (distribution, classant). Alors nous présentons dans des méthodes pour contrôler la structure logique des documents d'enchaînement (semistructure des données). En conclusion, nous présentons des méthodes de base pour traiter la sémantique des documents et des données, pour la recherche (recherche documentaire) et pour l'extraction de nouvelle information (exploitation de données).

Contenu:

Distributed data management: Fragmentation de base de données, gestion des données mobile, gestion des données de Peer-2-peer;
Semistructured Data Management: semistructuré Modèles de données, extraction de schéma et indexation, enchaînement sémantique;
Information Retrieval: Indexation des textes, recherche documentaire standard, moteurs de recherche du Web;
Data Mining : Exploitation de Règle d'Association, Classification, Groupement

Prérequis:

Bases de données relationnelles ou Introduction to information systems

Forme d'enseignement:

Ex cathedra + exercices

Bibliographie:

Notes de cours photocopiés

Objectives:

The lecture gives an overview of key problems in Web-based and mobile information management. It introduces in detail a selection of characteristic approaches, both from practice and research, and thus creates awareness for the specific challenges in distributed information management and typical solutions. The students will be able to identify the different problem classes in distributed information management (e.g. mobile data management) and corresponding techniques for solving them (e.g. indexing structures), to understand various standard methods in distributed information management (e.g. vector space information retrieval) and to apply these methods to (simple) practical problems. We proceed at increasing levels of abstraction. We start from the physical aspects of managing distributed and mobile data (distribution, indexing). Then we introduce into methods for managing the logical structure of Web documents (semistructured data). Finally, we introduce basic methods for dealing with the semantics of documents and data, both for search (information retrieval) and for the extraction of new information (data mining).

Content:

Distributed data management: Database fragmentation, Mobile data management, Peer-2-peer data management;
Semistructured Data Management: Semistructured data models, Schema extraction and indexing, Semantic Web;
Information Retrieval: Text indexing, Standard information retrieval, Web search engines
Data Mining: Association Rule Mining, Classification, Clustering

Required prior knowledge:

Bases de données relationnelles or Introduction to information systems

Form of teaching:

Ex cathedra + exercises

Matière examinée / subjects examined		Distributed information systems			
Session	PR1	Coefficient / Crédits ECTS	4	Forme de l'examen / Form of examination	Ecrit

Titre	Dynamical system theory for engineers
Title	Dynamical system theory for engineers

Enseignant(s) / Instructor(s)	De Feo Oscar: SC, Hasler Martin: SC		Langue / Language	EN
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours	
Informatique (2005-2006, Master semestre 1)	Cours: 4 Exercice: 2	1	opt	
Informatique (2005-2006, Master semestre 3)	Cours: 4 Exercice: 2	1	opt	
SC - EPFL (IS) (2005-2006, Master semestre 1)	Cours: 4 Exercice: 2	4	opt	
SC - EPFL (IS) (2005-2006, Master semestre 3)	Cours: 4 Exercice: 2	4	opt	

Objectifs:

L'étudiant sera capable de choisir les hypothèses et techniques de modélisation permettant d'établir un model mathématique (équations d'état) d'un phénomène décrit qualitativement. Pour les systèmes linéaires, il saura: prévoir les propriétés et résoudre des problèmes simples; de lien entre systèmes linéaires positifs et les problèmes statistiques. Dans le cas des systèmes non linéaires, il saura: distinguer, identifier, et analyser les différents comportements asymptotiques, y compris le comportement chaotique; esquisser et prédire le comportement qualitatif et déterminer leur stabilité locale et structurelle et de possibles applications pratiques.

Contenu:

Introduction: Systèmes dynamiques; Algèbre des schémas à blocs. **Systèmes Linéaires:** Définitions; Solution; Stabilité; Analyse géométrique; Variétés stables, instables et centraux; Contrôlabilité et observabilité; Identification ARMA LSQ; Systèmes positifs et probabilités; Matlab et l'analyse des systèmes linéaires. **Systèmes Fortement Non Linéaires:** Exemples; Invariantes génériques; Géométrie fractale; Non linéaires vs. linéaires; Comportement asymptotique; Basins d'attraction; Stabilité; Méthodes graphiques pour l'analyse et pour systèmes à faible dimension; Théorie ergodique; Stabilité structurelle et bifurcations (locales et globales); Perturbations singulières; Logiciels spécifiques.

Prérequis:

Circuits et systèmes I et II

Forme d'enseignement:

Ex cathedra et séances d'exercices, démos sur ordinateurs

Bibliographie:

O. De Feo & I. Belykh, Handsout, EPFL, Lausanne, Switzerland, 2004;
S. Strogatz, Nonlinear Dynamics and Chaos, Perseus, 1994

Objectives:

The student will be able of choosing the appropriate modeling techniques and hypothesis to establish a mathematical (state equations) model of a qualitatively described phenomenon. For the linear systems, he/she will know: how to anticipate their properties and to solve simple problems; about the link between positive linear systems and statistical problems. For nonlinear dynamical systems, he/she will know: how to distinguish, identify, and analyze the fundamentals different nonlinear behaviors, including chaotic behavior, sketch and predict their qualitative behavior and determine the local and structural stability and the practical applications concerning strongly nonlinear phenomena.

Content:

Introduction: Dynamical systems descriptions; Block schemes algebra. **Linear Systems:** Definitions; Solution; Stability; Geometrical analysis; Stable, unstable, and center manifolds; Reachability and observability; ARMA LSQ identification; Positive systems and probabilities; Matlab and the analysis of linear systems. **Strongly Nonlinear Systems:** Examples; Generic invariant sets; Fractal geometry; Linear vs. nonlinear systems; Asymptotic behavior and invariant sets stability; Basins of attraction; Stability; Graphical methods for the analysis; Low order methods; Ergodic theory; Structural stability and bifurcations; Local and Global bifurcations; Singular perturbations; Specific software.

Required prior knowledge:

Circuits et systèmes I et II

Form of teaching:

Ex cathedra and exercise sessions, demonstrations on computers

URLs		1) http://lanoswww.epfl.ch/studinfo/courses/cours_dynsys		
Matière examinée / subjects examined		Dynamical system theory for engineers		
Session	PRI	Coefficient / Crédits ECTS	7	Forme de l'examen / Form of examination
				Ecrit

Titre	Embedded systems
Title	Embedded systems

Enseignant(s) / Instructor(s)	Beuchat René: IN		Langue / Language	EN
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours	
Informatique (2005-2006, Master semestre 2)	Cours: 2 Projet: 2	4 6	opt	
Génie électrique et électronique (2005-2006, Master semestre 2)	Cours: 2 Projet: 2	1	opt	
SC - EPFL (IS) (2005-2006, Master semestre 2)	Cours: 2 Projet: 2	6	opt	
SC - EPFL (IS) (2005-2006, Master semestre 4)	Cours: 2 Projet: 2	6	opt	

Objectifs:

Ce cours, orienté matériel et interfaçage matériel, présente de façon détaillée les divers constituants d'un système embarqué. Une première partie décrit les divers constituants d'un système tels que les bus généraux parallèles et séries, les bus de processeurs asynchrones et synchrones, leurs caractéristiques communes et divergentes. Les mémoires complexes et leur interfaçage (DRAM, RAMBUS, DDR, etc...).

Les principes de processeurs embarqués sur FPGA hardcore et softcore sont étudiés et mis en oeuvre lors de laboratoires. La méthodologie de conception de tels systèmes est mise en application lors des travaux pratiques, notamment lors de conception d'interfaces programmables.

Des laboratoires sont associés pour les domaines principaux.

Contenu:

Bus synchrones et asynchrones, dynamique bus sizing
 Bus processeur, bus "backplanes"
 Bus série, USB, 1394, Ethernet
 Ecrans LCD, graphiques, caméras CMOS
 Organisation mémoire Little/big Endian
 Méthodologie et conception de systèmes embarqués
 Systèmes embarqués à FPGA, processeurs intégrés

Prérequis:

Introduction aux systèmes informatiques, Electronique, Systèmes logiques, Architecture des ordinateurs, Programmation (C/C++)

Préparation pour:

Systèmes embarqués en temps réel

Forme d'enseignement:

Ex cathedra et exercices

Remarque:

Liaison avec d'autres cours : Advanced Digital Design
 Informatique du temps réel

Objectives:

This course is oriented hardware and interfaces. It presents the different part of an embedded system.

The first part explain the different part of this kind of system, with standards parallel and serial bus, processor bus (asynchronous, synchronous) common and divergent characteristics and special memories.

FPGA hardcore and softcore embedded processors are described.

Conception methodology of some architecture is put in application with practical works.

Laboratories are associated with main topics.

Content:

Synchronous/asynchronous bus, dynamic bus sizing
 Processor bus, backplane bus
 Serial bus (USB, 1394, Ethernet)
 Basic on graphical screen and CMOS camera
 Memory organization, little/big endian
 Embedded systems conception
 FPGA embedded processor

Required prior knowledge:

Introduction aux systèmes informatiques, Electronique, Systèmes logiques, Architecture des ordinateurs, Programmation (C/C++)

Prerequisite for:

Real-time embedded systems

Form of teaching:

Ex cathedra and exercises

Matière examinée / subjects examined		Embedded systems			
Session	ETE	Coefficient / Crédits ECTS	4	Forme de l'examen / Form of examination	Oral

Titre	Foundations of image science
Title	Foundations of image science

Enseignant(s) / Instructor(s)	Fua Pascal: IN	Langue / Language	EN
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Informatique (2005-2006, Master semestre 1)	Cours: 2 Exercice: 1	3	opt
Informatique (2005-2006, Master semestre 3)	Cours: 2 Exercice: 1	3	opt
SC - EPFL (IS) (2005-2006, Master semestre 1)	Cours: 2 Exercice: 1	2	opt
SC - EPFL (IS) (2005-2006, Master semestre 3)	Cours: 2 Exercice: 1	2	opt

Objectifs:

Les cours avancés dans les domaines de la photographie digitale, de la vision par ordinateur et du graphique requièrent la maîtrise d'un certain nombre de techniques mathématiques et de leur implémentation.

Le but de ce cours est de donner aux étudiants cette maîtrise en combinant des cours ex-cathedra avec des travaux pratiques de développement et de prototypage sous JAVA.

Contenu:

Géométrie et radiométrie des images

- Caméras et géométrie projective
- Géométrie d'une ou plusieurs images
- Sources lumineuses, ombres et ombrage.
- Mesure et échantillonnage de l'intensité lumineuse.
- Couleur et texture

Filtrage et ses applications

- Filtres linéaires
- Convolution et séparabilité
- Transformée de Fourier
- Contours et texture

Optimisation discrète

- Programmation dynamique et chaînage de contours.
- Optimisation dans les graphes et segmentation.

Préparation pour:

Introduction to Computer Vision, Computer Graphics, Color Reproduction

Forme d'enseignement:

Ex cathedra, films, et exercices sur ordinateur

Forme du contrôle:

Avec contrôle continu

Bibliographie:

D. A. Forsyth, J. Ponce, Computer Vision: A Modern Approach, Prentice Hall, 2002.
R. Hartley and A. Zisserman, Multiple View Geometry in Computer Vision, Cambridge University Press, 2003.

Objectives:

Advanced classes in the fields of Digital Photography, Computer Vision, and Computer Graphics require the mastery of a certain number of mathematical techniques and of their actual implementations.

This course aims at supplying this knowledge by combining formal lectures and software development and prototyping exercises under JAVA.

Content:

Image Geometry and Radiometry

- Cameras and projective geometry
- Geometry of single and multiple images
- Light sources, shadows and shading
- Measuring and sampling light
- Color and Texture

Image Filtering and its Applications

- Linear Filters
- Convolution and separability
- Fourier Transform
- Edge and Texture Detection

Discrete Optimization

- Dynamic programming and edge linking
- Graph cuts and segmentation

Prerequisite for:

Introduction to Computer Vision, Computer Graphics, Color Reproduction

Form of teaching:

Ex cathedra, movies, and computer exercises

Form of examination:

With continuous control

Matière examinée / subjects examined		Foundations of image science			
Session	PRI	Coefficient / Crédits ECTS	4	Forme de l'examen / Form of examination	Ecrit

Titre	Genetic and developmental computing architectures
Title	Genetic and developmental computing architectures

Enseignant(s) / Instructor(s)	Ijspeert Auke: IN, Tempesti Gianluca: IN		Langue / Language	EN
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours	
Informatique (2005-2006, Master semestre 1)	Cours: 4 Exercice: 2		opt	
Informatique (2005-2006, Master semestre 3)	Cours: 4 Exercice: 2		opt	

Objectifs:

L'objectif principal de ce cours est celui de présenter à l'étudiant une vue d'ensemble de approches les plus courantes qui s'inspirent du monde de la biologie pour la conception des machines informatiques. La première partie de ce cours décrit le projet Embryonique, finalisé à établir un pont entre l'organisation des organismes vivants multicellulaires et l'architecture des ordinateurs (le design de réseaux cellulaires dotés de propriétés quasi-biologiques telles que l'autoréparation et l'autoreproduction). La deuxième partie s'inspire du processus d'évolution naturel pour suggérer des programmes et algorithmes génétiques. La troisième partie examine l'application des algorithmes génétiques et des réseaux neuronaux dans le domaine de la robotique, en portant une attention particulière au processus de locomotion et au domaine récent de la robotique modulaire. Des conférenciers externes, choisis parmi les spécialistes dans ces domaines, feront partie intégrante de ce cours

Contenu:

1. Embryonique
2. Automates et réseaux cellulaires auto-réplicateurs
3. Ontogenèse des êtres vivants
4. Réseaux cellulaires
5. Matrices de processeurs auto-réparatrices
6. Evolution naturelle et artificielle
7. Algorithmes génétiques
8. Logique floue
9. Robotique évolutionniste
10. Robotique modulaire

Prérequis:

Systèmes logiques, Architecture des ordinateurs I

Forme d'enseignement:

Cours avec exercices et laboratoire intégré. Cours en anglais avec des interventions en français.

Bibliographie:

"Bio-Inspired Computing Machines" (D. Mange, M. Tomassini), PPUR, Lausanne 1998

Objectives:

The primary objective of this course is to present the student with an overview of some of the most common approaches that draw inspiration from the world of biology in the design of computing machines. The first part of the course describes the Embryonics project, aimed at establishing a bridge between the organization of multi-cellular living organisms and the architecture of computer hardware (design of cellular networks endowed with quasi-biological properties such as self-repair and self-reproduction). The second part draws its inspiration from the evolutionary process in nature, describing analogous processes in computational media, the so-called genetic programs and algorithms. The third part examines in some detail the application of genetic algorithms and neural networks to the field of robotics, with particular attention to the process of locomotion and to the nascent field of modular robotics. External speakers, chosen among the specialists in their fields, are an integral part of the course.

Content:

1. Embryonics
2. Self-reproducing cellular automata and networks
3. Ontogenesis of living beings
4. Cellular networks
5. Self-repairing arrays
6. Natural and artificial evolution
7. Genetic algorithms
8. Fuzzy logic
9. Evolutionary robotics
10. Modular robotics

Required prior knowledge:

Systèmes logiques, Architecture des ordinateurs I

Form of teaching:

Course with exercises and integrated laboratory work. Course in English with some French lectures.

Matière examinée / subjects examined		Genetic and developmental computing architectures			
Session	PRI	Coefficient / Crédits ECTS	6	Forme de l'examen / Form of examination	Ctrl continu

Titre	Graphes et réseaux I
Title	Graphs and networks I

Enseignant(s) / Instructor(s)	de Werra Dominique: MA		Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours	
Informatique (2005-2006, Master semestre 1)	Cours: 2 Exercice: 2		opt	
Informatique (2005-2006, Master semestre 3)	Cours: 2 Exercice: 2		opt	
Mathématiques (2005-2006, Bachelor semestre 5)	Cours: 2 Exercice: 2		opt	
Enseignement maths - physique (2005-2006, Master semestre 1)	Cours: 2 Exercice: 2		opt	

Objectifs:

Familiariser l'étudiant avec l'utilisation des graphes comme instrument de modélisation dans les sciences de l'ingénieur, en informatique et en gestion.

Contenu:

Concepts de base de la théorie des graphes :
Chaînes et chemins, cycles et circuits, co-cycles et co-circuits, arbres et co-arbres.
Problèmes de connexité et de cheminement :
Arbres et arborescences optimaux, cycles et circuits eulériens ou hamiltoniens.
Flots dans les réseaux :
Algorithmes de flot maximum, de flot compatible, de flot à coût minimum. Construction de réseaux à performances optimales.
Diverses applications : problèmes d'ordonnement, carrés latins, etc.
Graphes planaires :
Algorithmes de reconnaissance, coloration des sommets/arêtes, graphe dual d'un graphe planaire.
Graphes parfaits :
Définitions et propriétés des graphes parfaits. Graphes triangulés, de permutation, d'intervalles, de comparabilité, parfaitement ordonnables, etc. Algorithmes de reconnaissance et d'optimisation combinatoire dans les graphes parfaits (coloration, stable maximum, etc.).
Algorithmes dans les graphes
Détermination du nombre chromatique et du nombre de stabilité d'un graphe quelconque. Variations et extensions. Bornes supérieures sur le nombre chromatique, bornes inférieures sur le nombre de stabilité et méthodes heuristiques.

Prérequis:

Algèbre linéaire, recherche opérationnelle

Préparation pour:

Modélisation de systèmes dans les science de l'ingénieur

Forme d'enseignement:

Cours ex cathedra, exercices en salle

Bibliographie:

R.K.Ahuja, T.L. Magnanti & J.B. Orlin : Network flows, Prentice-Hall, 1993.
D.B. West: Introduction to Graph Theory, Prentice-Hall, 2001.
D.de W., J.F. Hêche & T. Liebling: Recherche Opérationnelle pour ingénieurs I, PPUR 2003.

Objectives:

To show how graphs and their algorithms can be used for modelling and solving practical problems (e.g. in management and in computer science).

Content:

Basic concepts of graph theory :
Chains and paths, cycles and circuits, co-cycles and co-circuits, trees and co-trees.
Connectivity and routing problems:
Optimal trees and rooted trees, Eulerian or Hamiltonian cycles.
Network flows:
Algorithms for the maximum flow problem, the compatible flow problem, the minimum cost flow problem. Design of optimal networks. Applications to open shop scheduling, Latin squares, etc.
Planar graphs:
Recognition algorithms, edge/vertex, coloring dual of planar graphs.
Perfect graphs:
Definitions and properties of perfect graphs. Chordal graphs, interval graphs, permutation graphs, comparability graphs, perfectly orderable graphs, etc. Recognition algorithms, and algorithms for the solution of difficult combinatorial problems in perfect graphs (vertex coloring, maximum stable set, etc.).
Algorithms in graphs:
Computation of the chromatic number and the stability number of a graph.
Bounds on the chromatic number, and on the stability number.

Required prior knowledge:

Linear algebra, operational research

Prerequisite for:

Systems modelisation in the engineers sciences

Form of teaching:

Ex cathedra lecture, exercises in the classroom

Matière examinée / subjects examined		Graphes et réseaux I, II			
Session	ETE	Coefficient / Crédits ECTS	8	Forme de l'examen / Form of examination	Oral

Titre	Graphes et réseaux II
Title	Graphs and networks II

Enseignant(s) / Instructor(s)	de Werra Dominique: MA		Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours	
Informatique (2005-2006, Master semestre 2)	Cours: 2 Exercice: 2		opt	
Mathématiques (2005-2006, Bachelor semestre 6)	Cours: 2 Exercice: 2		opt	
Enseignement maths - physique (2005-2006, Master semestre 2)	Cours: 2 Exercice: 2		opt	

Objectifs:

Familiariser l'étudiant avec l'utilisation des graphes comme instrument de modélisation dans les sciences de l'ingénieur, en informatique et en gestion.

Contenu:

Concepts de base de la théorie des graphes :
Chaînes et chemins, cycles et circuits, co-cycles et co-circuits, arbres et co-arbres.
Problèmes de connexité et de cheminement :
Arbres et arborescences optimaux, cycles et circuits eulériens ou hamiltoniens.
Flots dans les réseaux :
Algorithmes de flot maximum, de flot compatible, de flot à coût minimum. Construction de réseaux à performances optimales.
Diverses applications : problèmes d'ordonnement, carrés latins, etc.
Graphes planaires :
Algorithmes de reconnaissance, coloration des sommets/arêtes, graphe dual d'un graphe planaire.
Graphes parfaits :
Définitions et propriétés des graphes parfaits. Graphes triangulés, de permutation, d'intervalles, de comparabilité, parfaitement ordonnables, etc. Algorithmes de reconnaissance et d'optimisation combinatoire dans les graphes parfaits (coloration, stable maximum, etc.).
Algorithmes dans les graphes
Détermination du nombre chromatique et du nombre de stabilité d'un graphe quelconque. Variations et extensions. Bornes supérieures sur le nombre chromatique, bornes inférieures sur le nombre de stabilité et méthodes heuristiques.

Prérequis:

Algèbre linéaire, recherche opérationnelle

Préparation pour:

Modélisation de systèmes dans les science de l'ingénieur

Forme d'enseignement:

Cours ex cathedra, exercices en salle

Bibliographie:

R.K.Ahuja, T.L. Magnanti & J.B. Orlin : Network flows, Prentice-Hall, 1993.
D.B. West: Introduction to Graph Theory, Prentice-Hall, 2001.
D.de W., J.F. Hêche & T. Liebling: Recherche Opérationnelle pour ingénieurs I, PPUR 2003.

Objectives:

To show how graphs and their algorithms can be used for modelling and solving practical problems (e.g. in management and in computer science).

Content:

Basic concepts of graph theory :
Chains and paths, cycles and circuits, co-cycles and co-circuits, trees and co-trees.
Connectivity and routing problems:
Optimal trees and rooted trees, Eulerian or Hamiltonian cycles.
Network flows:
Algorithms for the maximum flow problem, the compatible flow problem, the minimum cost flow problem. Design of optimal networks. Applications to open shop scheduling, Latin squares, etc.
Planar graphs:
Recognition algorithms, edge/vertex, coloring dual of planar graphs.
Perfect graphs:
Definitions and properties of perfect graphs. Chordal graphs, interval graphs, permutation graphs, comparability graphs, perfectly orderable graphs, etc. Recognition algorithms, and algorithms for the solution of difficult combinatorial problems in perfect graphs (vertex coloring, maximum stable set, etc.).
Algorithms in graphs:
Computation of the chromatic number and the stability number of a graph.
Bounds on the chromatic number, and on the stability number.

Required prior knowledge:

Linear algebra, operational research

Prerequisite for:

Systems modelisation in the engineers sciences

Form of teaching:

Ex cathedra lecture, exercises in the classroom

Matière examinée / subjects examined		Graphes et réseaux I, II			
Session	ETE	Coefficient / Crédits ECTS	8	Forme de l'examen / Form of examination	Oral

Titre	Human computer interaction
Title	Human computer interaction

Enseignant(s) / Instructor(s)	Pu Faltings Pearl: IN		Langue / Language	EN
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours	
Informatique (2005-2006, Master semestre 2)	Cours: 2 Exercice: 1	5	opt	
SC - EPFL (IS) (2005-2006, Master semestre 2)	Cours: 2 Exercice: 1	5	opt	
SC - EPFL (IS) (2005-2006, Master semestre 4)	Cours: 2 Exercice: 1	5	opt	

Objectifs:

Concevoir de façon créative des produits et service IT en tenant compte de l'utilisation est difficile. C'est un compromis entre l'application de technologies intelligentes et le maintien de la simplicité d'emploi. C'est avec cet objectif de "design to compel" que le cours enseigne les concepts d'ergonomie et de l'Interaction Homme-Machine. L'enseignement est souligné par 3 à 4 projets de conception avec le but de "libérer" l'énergie créative des étudiants, et les rendre capable d'établir et atteindre les objectives de « usability ». Les projets couvrent la conception, le prototypage et les tests de logiciels interactifs. Java, ou des outils tels que JavaScript ou Macromedia Director, sont nécessaires pour les prototypage.

Contenu:

- Concepts de base de l'interaction homme-machine
 - Caractéristique humaines
 - "Erreurs" humaines
 - Utilisabilité vs. interfaces conviviales
 - Le principe KISS
 - Techniques de brainstorming
 - Conception et prototypage pour l'utilisabilité
 - Test d'utilisabilité
- Les sujets avancés de l'Interaction Homme-Machine suivants seront abordés au long du cours :
- Visualisation de l'information
 - Agents personnels intelligents
 - Traitements dépendants du contexte

Prérequis:

Programming course, basic knowledge of human computer interaction theory

Forme d'enseignement:

Lectures, case studies, group projects

Remarque:

Liaison avec d'autres cours : Software engineering course; conceptual design of databases

Bibliographie:

Teaching notes and suggested reading material

Objectives:

Creative design of compelling IT products and services with usability in mind is hard. It's a compromise between providing smart technology, while keeping the software easy to use. Under such a "design to compel" objective, the course teaches students concepts of ergonomics and human computer interaction by guiding them through a set of 3 to 4 design projects that intend to "unlock" their creative energy and enable them to define and execute usability objectives. The projects range from designing, prototyping, and testing interactive software. Java, or a tool such as JavaScript, Macromedia Director, is necessary.

Content:

- Basic concepts of human-computer interaction
- Human characteristics
- Human "errors"
- Usability vs. user friendly interfaces
- KISS principle
- Brainstorming techniques
- Design and prototyping for usability
- Usability testing

The following advanced topics in human computer interaction will be presented throughout the course:

- Information visualization
- Intelligent and personal agents
- Context-aware computing

Required prior knowledge:

Programming course, basic knowledge of human computer interaction theory

Form of teaching:

Lectures, case studies, group projects

Note:

Connection with other course : Software engineering course; conceptual design of databases

Matière examinée / subjects examined		Human computer interaction			
Session	ETE	Coefficient / Crédits ECTS	4	Forme de l'examen / Form of examination	Ctrl continu

Titre	Industrial automation
Title	Industrial automation

Enseignant(s) / Instructor(s)	Kirmann Hubert: SC		Langue / Language	EN
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours	
Informatique (2005-2006, Master semestre 2)	Cours: 2 Projet: 1	4	opt	
Génie électrique et électronique (2005-2006, Master semestre 2)	Cours: 2 Projet: 1		opt	
Systèmes de communication (2005-2006, Bachelor semestre 6)	Cours: 2 Projet: 1		opt	

Objectifs:

Ce cours s'adresse aux informaticiens, électroniciens ou ingénieurs de communication concevant ou appliquant des systèmes d'automation, depuis les petits laboratoires jusqu'aux grandes usines. L'Automation Industrielle concerne les moyens de calcul et de communication conduisant usines, centrales et réseaux électriques, véhicules et autres systèmes embarqués. Elle englobe toute la hiérarchie de contrôle-commande depuis les capteurs de mesure, en passant par les automates, les bus de communication, la visualisation, l'archivage jusqu'à la gestion de production et des ressources de l'entreprise. Ce cours pratique n'exige pas comme préalable la théorie du contrôle automatique. Il complète les cours de téléinformatique avec l'accent sur l'usage industriel. Il comporte des laboratoires sur des systèmes réels et des visites d'usine.

Contenu:

1. Processus et usines, architecture de contrôle-commande
2. Instrumentation, Contrôle et Automates (AP)
3. Réseaux de communication industriels, bus de terrain
4. Protocoles pour dispositifs (HART, MMS) et OPC
5. Interface homme-machine, fonctions SCADA
6. Gestion de production, production par lots (ISA88, 95)
7. Configuration, test et mise en service
8. Temps réel et évaluation des besoins en performances
9. Tolérance aux fautes et sûreté, analyse et calcul

Prérequis:

Réseaux de communication

Forme d'enseignement:

Orale, exercices, travaux pratiques

Bibliographie:

Nussbaumer, Informatique Industrielle

Objectives:

This course is directed to the informatics, electronics or communication engineers who design or apply industrial automation systems, from small laboratories to large enterprises. Industrial Automation considers the computer and communication systems that control factories, energy production and distribution, vehicles and other embedded systems. Industrial Automation encompasses the whole control hierarchy from sensors, motors, controllers, communication busses, operator visualisation, archiving and up to manufacturing execution systems and enterprise resource management. This course is application-oriented and does not require previous knowledge in control theory. It complements communication systems courses with a focus on industrial application. It includes workshops giving hands-on experience and factory visits.

Content:

1. Processes and plants, control system architecture
2. Instrumentation, Control and Controllers (PLC)
3. Industrial communication networks, field busses
4. Device access protocols (HART, MMS and OPC)
5. Human-Machine Interface, SCADA functions
6. Manufacturing Execution Systems, Batch (ISA 88, 95)
7. Engineering, Commissioning and Test
8. Real-time response and performance requirement analysis
9. Fault-tolerance and safety, analysis and computation

Form of teaching:

Oral, exercises, practical work

Matière examinée / subjects examined		Industrial automation			
Session	ETE	Coefficient / Crédits ECTS	3	Forme de l'examen / Form of examination	Oral

Titre	Intelligent Agents
Title	Intelligent agents

Enseignant(s) / Instructor(s)	Faltings Boi: IN	Langue / Language	EN
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Informatique (2005-2006, Master semestre 1)	Cours: 3 Exercice: 3	4 5	opt
Informatique (2005-2006, Master semestre 3)	Cours: 3 Exercice: 3	4 5	opt
SC - EPFL (IS) (2005-2006, Master semestre 1)	Cours: 3 Exercice: 3	5	opt
SC - EPFL (IS) (2005-2006, Master semestre 3)	Cours: 3 Exercice: 3	5	opt

Objectifs:

Les agents intelligents sont une nouvelle technologie pour l'implémentation efficace de grands systèmes logiciels, centralisés ou distribués. Ils trouvent de plus en plus d'applications dans divers domaines comme les systèmes d'information et le commerce électronique.

L'objectif de ce cours est d'apprendre les technologies pour l'implémentation d'agents intelligents et de systèmes multi-agents ainsi que les théories sous-jacentes.

Contenu:

Le cours traite 4 thèmes principaux:

- 1) Agents simples:
Algorithmes pour des programmes de jeux, agents réactifs, reinforcement learning, modèles logiques d'agents
- 2) Agents rationnels:
Planification automatique, algorithmes distribués pour la satisfaction de contraintes, coordination d'agents
- 3) Sémantique Web:
Plateformes d'agents, utilisation d'ontologies, standards pour les web services
- 4) Agents économiques:
Théorie des jeux, principes de la négociation et d'économies électroniques.

Prérequis:

Intelligence artificielle

Forme du contrôle:

avec contrôle continu

Bibliographie:

Divers papiers techniques en langue anglaise

Objectives:

Intelligent agents are a new technology for efficiently implementing large software systems which may also be distributed. They are increasingly applied to problems ranging from information systems to electronic commerce.

This course teaches students the main technologies for implementing intelligent agents and multi-agent systems as well as their underlying theories.

Content:

The course contains 4 main subject areas:

- 1) Basic models and algorithms for agents:
game-playing algorithms, reactive agents and reinforcement learning, logical (BDI) agent models.
- 2) Rational agents:
Models and algorithms for rational, goal-oriented behavior in agents: planning, distributed algorithms for constraint satisfaction, coordination techniques for multi-agent systems.
- 3) Semantic Web:
Agent platforms, ontologies and markup languages, web services and standards for their definition and indexing.
- 4) Self-interested agents:
Models and algorithms for implementing self-interested agents motivated by economic principles: relevant elements of game theory, models and algorithms for automated negotiation, electronic auctions and marketplaces.

Required prior knowledge:

Intelligence artificielle

Form of examination:

with continuous control

Matière examinée / subjects examined		Intelligent Agents			
Session	PRI	Coefficient / Crédits ECTS	6	Forme de l'examen / Form of examination	Ecrit

Titre	Mobile networks
Title	Mobile networks

Enseignant(s) / Instructor(s)	Hubaux Jean-Pierre: SC		Langue / Language	EN
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours	
Informatique (2005-2006, Master semestre 1)	Cours: 2 Exercice: 1	5	opt	
Informatique (2005-2006, Master semestre 3)	Cours: 2 Exercice: 1	5	opt	
SC - EPFL (IS) (2005-2006, Master semestre 1)	Cours: 2 Exercice: 1	1 3 5 7	obl	
SC - EPFL (IS) (2005-2006, Master semestre 3)	Cours: 2 Exercice: 1	1 3 5 7	obl	

Objectifs:

Un premier objectif de ce cours est de fournir une compréhension détaillée des techniques permettant de supporter la mobilité dans les réseaux sans fil (au-dessus de la couche physique): accès multiple, gestion de la mobilité, hand-over, roaming, sécurité, et planification de réseau (y compris l'estimation de la capacité). Un deuxième objectif est d'illustrer ces techniques en montrant leur usage dans les réseaux mobiles les plus courants, à savoir les réseaux cellulaires et les réseaux locaux sans fil. Un troisième objectif consiste à fournir une introduction aux réseaux de capteurs.

Contenu:

- Introduction: réseaux sans fil et mobilité
- Techniques d'accès multiple sur un canal radio
- Rappels sur la sécurité
- Principes de fonctionnement des réseaux locaux sans fil; un exemple important: IEEE 802.11
- Exercices pratiques sur IEEE 802.11; illustration des vulnérabilités et des contre-mesures
- Hotspots WiFi: défis techniques et solutions possibles
- Mobilité dans les réseaux IP; Mobile IPv4 et v6
- Réseaux cellulaires: capacité, gestion de la mobilité; hand-over; roaming; sécurité; facturation
- Exemples de réseaux cellulaires: GSM, GPRS, et UMTS
- Introduction aux réseaux de capteurs.

Prérequis:

Introduction aux réseaux de communications ou équivalent

Forme d'enseignement:

Ex cathedra et exercices en salle

Bibliographie:

Handouts

Objectives:

A first objective of this course is to provide a deep understanding of the techniques used to support mobility in wireless networks (above the physical layer): multiple access, mobility management, hand-over, roaming, security, and network planning (including capacity estimation). A second objective is to illustrate these techniques by showing their usage in the most relevant mobile networks, namely cellular networks and wireless Local Area Networks. A third objective is to provide an introduction to sensor networks.

Content:

- Introduction: wireless and mobility
- Multiple access techniques over a radio channel
- Reminders on security
- Operating principles of wireless LANs; a prominent example: IEEE 802.11
- Hands-on exercises on IEEE 802.11; illustration of vulnerabilities and counter-measures
- Wi-Fi hotspots: technical challenges and possible solutions
- Mobility in IP networks; Mobile IPv4 and v6
- Cellular networks: capacity; mobility management; hand-over; roaming; security; billing
- Examples of cellular networks: GSM, GPRS and UMTS
- Introduction to sensor networks

Required prior knowledge:

Introduction aux réseaux de communications ou équivalent

Form of teaching:

Ex cathedra and exercises in class

Matière examinée / subjects examined		Mobile networks			
Session	PRI	Coefficient / Crédits ECTS	4	Forme de l'examen / Form of examination	Ecrit

Titre	Modelling the immune system
Title	Modelling the immune system

Enseignant(s) / Instructor(s)	Kraehenbuehl Jean-Pierre: SC, Le Boudec Jean-Yves: SC, Martinoli Alcherio: SC	Langue / Language	EN
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc / filière / orient	Type
Informatique (2005-2006, Master semestre 2)	Cours:1 Exercice:1	1	opt
SC - EPFL (IS) (2005-2006, Master semestre 2)	Cours:1 Exercice:1	4	opt
SC - EPFL (IS) (2005-2006, Master semestre 4)	Cours:1 Exercice:1	4	opt

Objectifs:

Comprendre le système immunitaire pour le modéliser. Connaître les outils de modélisation correspondant.

Contenu:

Le cours est donné en anglais. Les termes techniques sont donnés en anglais seulement pour éviter des confusions.
 Dans une première partie, on décrit qualitativement le fonctionnement global du système immunitaire humain en utilisant un logiciel d'enseignement par ordinateur.
 Dans une seconde partie, on étudie les méthodes de modélisations quantitatives susceptibles d'être utiles en biologie.
 Dans une troisième partie, on effectue des études de cas.
 Le cours est sous la forme de séminaires, et est basé sur des papiers de recherche récents.

Prérequis:

Cours de base en biologie + en probabilité

Préparation pour:

Recherche en biologie mathématique

Forme d'enseignement:

Ex cathedra et exercices assistés

Objectives:

To understand the Immune System (IS) in order to model it. To acquire knowledge about the tools required to model the immune system

Content:

I. Introduction to the IS: individual work on the web site (e-learning) and seminars with specific questions related to the immune system
 1. Overview
 2. The innate IS - Cells : Monocytes, Macrophages, Dendritic cells, Neutrophils, Eosinophils, Basophils, Mast cells , Natural killer cells; Molecules: Acute-phase proteins, Complement, Cytokines, Chemokines; Innate response: Recognition, Effector mechanisms, Inflammatory response, Inflammation, Phagocytosis, Cytolysis
 3. The adaptive IS - Cells: T-lymphocytes, B-lymphocytes; Molecules: Antibodies, Cytokines, Chemokines, Interleukins, Interferons; Organs: primary lymphoid organs (bone marrow and thymus), secondary lymphoid organs (spleen, lymph nodes, MALT B1_Adaptive_organsII), tertiary lymphoid organs; Adaptive response: recognition, MHC molecules, antigen presentation, clonal expansion, immune response versus tolerance, effector mechanisms.
 4. Diseases of the IS - Allergy, autoimmunity, immunodeficiencies and AIDS, tumoral immunity
II. Modelling Toolbox
 Microscopic models (stochastic cellular automata, multi-agent models); macroscopic models (difference and differential equations); simulators of the Immune System; scaling methods (stage-structured populations)
III. Case Studies
 Rheumatoid factor paradox; immune recognition during HIV; immunology of tumor growth

Required prior knowledge:

Cours de base en biologie + en probabilité

Prerequisite for:

Research in mathematical biology

Form of teaching:

Ex cathedra and aided exercises

Bibliographie:

Immunology Online: <http://www.iol.ch/>; Charles Janeway, Paul Travers, Mark Walport , Mark Schlomchik. "Immunobiology: The immune system in health and disease". 6th edition Garland Science. Churchill Livingstone; Lauren Sompayrac, "How the Immune System Works", 2nd edition, Blackwell Publishing; Richard A. Goldsby, Thomas J. Kindt and Barbara A. Osborne, "Kuby Immunology" (4th ed), W.H. Freeman; M. Mitchell "Computation in cellular automata: A selected review", In Non standard Computation; Burton Voorhees and Catherine Beauchemin. Point Mutations and Transitions Between Cellular Automata Attractor Basins. arXiv, nlin.CG/0306033, June 17, 2003.; D. L. Chao, M. P. Davenport, S. Forrest, and A. S. Perelson. "Stochastic Stage-structured Modeling of the Adaptive Immune System". Proceedings of the IEEE Computer Society Bioinformatics Conference (CSB 2003), pp 124-131. IEEE Press, Los Alamitos, California, 2003; Fernando Esponda and Elena S. Ackley and Stephanie Forrest and Paul Helman, "On-line Negative Databases"; Reading assignments available online

Matière examinée / subjects examined		Modelling the immune system			
Session	ETE	Coefficient / Crédits ECTS	3	Forme de l'examen / Form of examination	Oral

Titre	Models of biological sensory-motor systems
Title	Models of biological sensory-motor systems

Enseignant(s) / Instructor(s)	Ijspeert Auke: IN		Langue / Language	EN
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours	
Informatique (2005-2006, Master semestre 1)	Cours: 2 Exercice: 2	1	opt	
Informatique (2005-2006, Master semestre 3)	Cours: 2 Exercice: 2	1	opt	
SC - EPFL (IS) (2005-2006, Master semestre 1)	Cours: 2 Exercice: 2	4	opt	
SC - EPFL (IS) (2005-2006, Master semestre 3)	Cours: 2 Exercice: 2	4	opt	

Objectifs:

- (1) Revue de différents types de modèles numériques du contrôle de la locomotion et du mouvement, de la coordination sensori- motrice, de la perception, et de l'apprentissage chez l'animal
- (2) Présentation des différents types de techniques utilisées dans le cadre de ces modèles
- (3) Analyse de comment ces modèles et ces techniques peuvent être utilisés en informatique, dans des domaines tels que la robotique, la vision par ordinateur, et l'interaction homme- ordinateur.

Contenu:

Les points suivants seront adressés :

Modèles numériques de systèmes moteurs : modèles à base de réseaux de neurones du contrôle de la locomotion chez les invertébrés et les vertébrés, génération de rythmes à l'aide de « central pattern generators », réflexes, contrôle de l'équilibre, contrôle des membres supérieurs, « force fields », modèles internes pour le contrôle du mouvement (cinématique inverse et dynamique inverse), génération de mouvements complexes, coordination sensori-motrice, apprentissage moteur, application aux robots à pattes et robots humanoïdes, comparaison avec les techniques de contrôle traditionnelles en robotique

Modèles numériques de systèmes sensoriels : différents types de yeux, traitement visuel dans la rétine, vagues pour traitement d'images, systèmes visuels de la salamandre et du primate, voies du « où » et du « quoi », saccades, mécanismes d'attention, traitement du son et autres modalités sensorielles, fusion sensorielle, apprentissage, application à la vision par ordinateur, la robotique et l'interaction homme-ordinateur, comparaison avec des algorithmes traditionnels de traitements d'images et d'autres modalités sensorielles.

Le cours impliquera également des travaux pratiques au cours desquels les étudiants développeront leurs propres simulations de systèmes sensori-moteurs.

Forme d'enseignement:

Ex cathedra

Objectives:

- (1) To provide a comprehensive overview of numerical models developed for studying locomotion and movement control, sensory-motor coordination, perception, and learning in animals
- (2) To present different types of techniques used in such types of modeling
- (3) To analyze how these models and techniques can be used in computer science, in fields such as robotics, machine vision, and human-computer interaction.

Content:

Topics that will be addressed include:

Numerical models of motor systems : neural network models of control of locomotion in invertebrates and vertebrates, rhythm generation in central pattern generators, reflexes, control of balance, control of upper limbs, force fields, internal models for movement control (inverse kinematics and inverodynamics), generation of complex movements, sensory-motor coordination, motor learning, applications to legged and humanoid robots, comparison with traditional control techniques in robotics

Numerical models of sensory systems : different types of eyes, visual processing in the retina, wavelets for visual processing, salamander and primate visual systems, the « where » and « what » pathways, saccades, attentional mechanisms, processing of sound and other sensory modalities, sensory fusion, learning, applications to machine vision, robotics, and human-computer interaction, comparison with traditional sensory processing algorithms

The course will also involve practicals in which students will develop their own numerical simulations of sensory-motor systems.

Form of teaching:

Ex cathedra

Matière examinée / subjects examined		Models of biological sensory-motor systems			
Session	PRI	Coefficient / Crédits ECTS	4	Forme de l'examen / Form of examination	Ecrit

Titre	Multimedia documents
Title	Multimedia documents

Enseignant(s) / Instructor(s)	Vanoirbeek Christine: IN		Langue / Language	EN
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours	
Informatique (2005-2006, Master semestre 2)	Cours: 4 Exercice: 2	5	opt	
SC - EPFL (IS) (2005-2006, Master semestre 2)	Cours: 4 Exercice: 2	5	opt	
SC - EPFL (IS) (2005-2006, Master semestre 4)	Cours: 4 Exercice: 2	5	opt	

Objectifs:

Les systèmes d'informations actuels, en particulier pour une exploitation collaborative à travers la plateforme WWW, reposent sur l'utilisation croissante de documents multimédia. Le cours a pour objectif de décrire les modèles de représentation et les méthodes de traitement spécifiques à de tels systèmes. Il présente et discute les solutions actuelles (et émergentes) apportées par les normes pour répondre aux problèmes d'échange, d'interopérabilité et de mise en oeuvre d'applications qui reposent sur le concept de documents multimédia. Il couvre en particulier les techniques utilisées pour l'analyse et l'indexation de documents multimédia et démontre leur utilité dans le contexte de la recherche d'information

Contenu:

Les bases théoriques seront enseignées pour décrire les modèles dont découlent les normes de représentation structurée des documents

- Représentation des différentes structures de documents: structuration logique (XML), physique (CSS, XSL) et hypertexte (HTML, HyTime, Xlink, etc.).
- Représentation des documents composites et technologie multimédia: standards et méthodes de compression (JPEG, MPEG), documents actifs (JAVA), documents en temps que composants logiciels.
- Techniques de traitement et de transformations de structures de documents.
- Analyse et indexation de documents multimédia (sons, images, vidéo).

Forme d'enseignement:

Ex cathedra et exercices

Forme du contrôle:

Avec contrôle continu

Objectives:

Modern information systems, especially dedicated to the WWW environment, increasingly rely on multimedia documents. The goal of this course is to describe the models of representation and the processing methods that those systems use. The solutions offered by the developing standards of multimedia components to the problems of document exchange and interoperability, and multimedia document platforms will be presented and discussed. Techniques used in the analysis of multimedia documents will be covered, and their usefulness will be shown in the development of indexation and classification methods for information retrieval.

Content:

The theoretical foundations of models and standards for representing structured documents will be taught.

- Representation methods for structured documents: logical structure (XML), physical structures (CSS, XSL), and Hypertext (HTML, HyTime, Xlink, etc.).
- Representation of composite documents and multimedia technology: image and video compression techniques (JPEG, MPEG), active documents (JAVA), documents as software components.
- Management and transformation of structured documents.
- Component analysis and indexing (sound, images and video)

Form of teaching:

Ex cathedra and exercises

Form of examination:

With continuous control

Matière examinée / subjects examined		Multimedia documents			
Session	ETE	Coefficient / Crédits ECTS	6	Forme de l'examen / Form of examination	Ecrit

Titre	Parallélisation de programmes sur grappes de PC
Title	Program parallization on PC clusters

Enseignant(s) / Instructor(s)	Hersch Roger: IN	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Informatique (2005-2006, Master semestre 1)	Cours: 1 Projet: 2		opt
Informatique (2005-2006, Master semestre 3)	Cours: 1 Projet: 2		opt

Objectifs:

De plus en plus d'applications exigent une puissance de calcul et des débits d'entrées-sorties qui ne peuvent être offerts que par plusieurs ordinateurs travaillant simultanément. Ce cours vise à introduire les problèmes et méthodes pour la programmation parallèle sur grappes de PC

Contenu:

Contenu du cours:

- Architectures parallèles
- Méthodes de parallélisation,
- Métriques de performances,
- Modélisation des performances,
- Développement de programmes parallèles,
- Débusquage d'erreurs,
- Mesure des temps d'exécution,
- Contrôle de flux et équilibrage de charges

Environnement de développement:

- Visual C++ sous Windows, Sun Unix ou Linux
- Librairie DPS pour la création d'ordonnancements parallèles

Mini-projet:

Choix d'un problème, analyse, prédiction du gain de performances, développement du programme, test et comparaison avec les performances prédites

Projets proposés: algorithmes de tri, satisfaction de clauses booléennes, tour du cheval, décryptage de message, voyageur du commerce, traitement d'image, assemblage de puzzle, Transformée de Fourier rapide, apprentissage non-supervisé, systèmes d'équations linéaires, corps célestes (N-Body), transformée de Hough, automates cellulaires.

Forme d'enseignement:

Cours, laboratoire et mini-projet (C, C++)

Bibliographie:

Cours polycopié: Program Parallelization, site DPS : <http://dps.epfl.ch>

Objectives:

Demanding applications may require the processing power and/or I/O throughput offered by multiple PCs connected by Fast or Gigabit Ethernet. The course will introduce the problems and methods of program parallelization on PC clusters

Content:

Content:

- parallel architectures,
- parallelization methods,
- multi-threaded parallel programming
- parallelization metrics,
- theoretical performance models,
- parallel program development,
- debugging techniques and
- measurement of program execution times
- flow control & load balancing

Environment:

- Visual C++ under Windows, Sun Unix or Linux
- DPS C++ library for creating flowgraphs defining parallel execution schedules.

Project: Select a problem, predict the speedup, develop the parallel program (1 to 8 PC's) and compare predicted and measured performances.

Proposed projects: mergesort, bucket sort, satisfaction of boolean clauses, knight tour, descrypting of messages encrypted by permutation, travelling salesman, zooming in color image, monkey puzzle, FFT, creation of a color lookup table by unsupervised learning, linear equation systems (Jacobi iterations, Gaussian elimination), N-Body, Hough transform, LU decomposition, cellular automaton (image skeletonization).

Form of teaching:

Course, laboratory and short-project

URLs	1) http://dps.epfl.ch				
Matière examinée / subjects examined	Parallélisation de programmes sur grappes de PC				
Session	PRI	Coefficient / Crédits ECTS	3	Forme de l'examen / Form of examination	Ctrl continu

Titre	Pattern classification and machine learning
Title	Pattern classification and machine learning

Enseignant(s) / Instructor(s)	Gerstner Wulfram: IN, Hasler Martin: SC	Langue / Language	EN
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Informatique (2005-2006, Master semestre 2)	Cours: 4 Exercice: 2	1 3	opt
SC - EPFL (IS) (2005-2006, Master semestre 2)	Cours: 4 Exercice: 2	4	opt
SC - EPFL (IS) (2005-2006, Master semestre 4)	Cours: 4 Exercice: 2	4	opt

Objectifs:

La classification de données (images, textes, sons) est une tâche qui est à la base de toute apprentissage et reconnaissance automatique. L'objectif du cours est la maîtrise des algorithmes de classification, en particulier les réseaux de neurones artificiels, les méthodes classiques basées sur la règle de Bayes, les méthodes modernes basées sur les vecteurs à support ainsi que la compréhension de la théorie statistique de l'apprentissage.

Contenu:

I. Introduction: Classification et apprentissage supervisé

- Le problème d'une classification automatique des données

II. Réseaux de neurones artificiels

- Perceptron simple et séparabilité linéaire
- Réseaux multicouches et l'algorithme BackProp
- Le problème de la généralisation
- Applications

III. Décisions optimales et estimation de densité

- Maximum likelihood et Bayes
- Mixture Models et l'algorithme EM

IV. Comparaison de réseaux de neurones et méthodes classiques

- Réseaux RBF et logique flou
- Introduction au « Support vector machines »

V. Théorie statistique de l'apprentissage

- Introduction informelle
- Définition du problème d'apprentissage statistique
- Minimisation du risque empirique
- Dimension VC (Vapnik - Chervonenkis)
- Formalisation des « Support vector machines »

Prérequis:

Probabilité et statistique I, II ; Analyse I, II, III, et Programmation I

Forme d'enseignement:

Ex cathedra. Exercices en salle et sur ordinateur, miniprojet

Forme du contrôle:

Avec contrôle continu

Bibliographie:

Polycopiés : Réseau de Neurones Artificiels, Statistical theory of learning; Exercices et Initiation : Neural JAVA ; C. Bishop : Neural Networks for Pattern Recognition, Oxford, 1995 ; S. Haykin : Neural Networks, Prentice Hall, 1994 ; V. Vapnik : The Nature of Statistical Learning Theory, Springer, 1995

Objectives:

Data classification is at the heart of all learning and recognition. In this course the student will learn to master all relevant algorithms (artificial neural networks, Bayes classification, support vector machine) and understand the fundamentals of statistical learning theory.

Content:

I. Introduction: Classification and supervised learning

- The problem of automatic classification

II. Artificial Neural Networks

- Simple perceptrons and linear separability
- Multilayer Perceptrons: Backpropagation Algorithm
- The problem of generalization
- Applications

III. Optimal decision boundary and density estimation

- Maximum Likelihood and Bayes
- Mixture Models and EM-algorithm

IV. Comparison of classical and modern methods

- Network RBF and fuzzy logic
- Introduction to « Support vector machines »

V. Statistical learning theory

- Informal introduction
- Definition of the statistical learning problem
- Empirical risk minimization
- VC-dimension (Vapnik - Chervonenkis)
- « Support vector machines » and learning theory

Required prior knowledge:

Probabilité et statistique I, II ; Analyse I, II, III, et Programmation I

Form of teaching:

Ex cathedra. Exercises in class and on the computer, miniprojet

Form of examination:

With continuous control

Matière examinée / subjects examined		Pattern classification and machine learning			
Session	ETE	Coefficient / Crédits ECTS	6	Forme de l'examen / Form of examination	Ecrit

Titre	Performance evaluation
Title	Performance evaluation

Enseignant(s) / Instructor(s)	Le Boudec Jean-Yves: SC		Langue / Language	EN
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours	
Informatique (2005-2006, Master semestre 2)	Cours: 4 Exercice: 2	2 4 5	opt	
SC - EPFL (IS) (2005-2006, Master semestre 2)	Cours: 4 Exercice: 2	3 5	opt	
SC - EPFL (IS) (2005-2006, Master semestre 4)	Cours: 4 Exercice: 2	3 5	opt	

Objectifs:

L'évaluation de performance est souvent la partie critique d'un projet de recherche en informatique ou système de communication. Il est souvent malaisé de répondre à des questions telles que :

- dois-je faire une analyse de file d'attente complexe ou bien est-il possible d'estimer rapidement la performance attendue de manière approximative ?
- combien de temps dois-je faire tourner ma simulation ?
- dois-je éliminer le début de ma simulation pour que le système se stabilise et pourquoi ?
- je simule un modèle de mobilité mais la vitesse moyenne ne correspond pas à mes attentes; pourquoi ?
- qu'est ce qu'un intervalle de confiance ? comment les obtenir ?

Le but de ce cours est de répondre à ces questions, et bien d'autres encore, en bref, d'acquérir les éléments de base indispensables à l'évaluation de performance d'un projet informatique ou de communications (fondements théoriques et pratiques).

Contenu:

Méthodes. Une checklist pour l'évaluation de performance. La méthode scientifique. Les principes de Dijkstra et Occam.
Théorie et Pratique de la Simulation. Que se passe-t-il dans une simulateur ? Stationarité et ergodicité. Les points de vue temporels et événementiels. Le calcul de Palm. Simuler des distributions. Simulation parfaite. **Un Peu de Statistique.** Intervalles de confiance, méthodes exactes et asymptotiques. Tests. Analyse factorielle. **Les Files d'Attente Pour Ceux Qui n'ont Pas le Temps.** Lois opérationnelles, Little et lois des flux forcés. Analyse par bottleneck. Analyse transitoire. Non-linéarités. **Phénomènes de Performance.** Patterns et paradoxes. **Modélisation de la Charge.** Self similarité. Prédiction de charge. La méthode de Box-Jenkins. **Laboratoires.** Utiliser un outil de calcul statistique (Matlab ou S-PLUS). Mesures. Le générateurs de charge SURGE. Le simulateur ns2.

Prérequis:

Cours de base en probabilité

Forme d'enseignement:

Cours, exercices labos, devoirs

Bibliographie:

« Performance Evaluation », Notes de cours, Jean-Yves Le Boudec

Objectives:

Performance Evaluation is often the critical part in a research project in computer or communication systems. It is often difficult to address questions like

- Is it possible to quickly estimate the performance of my system without performing a detailed queuing analysis ?
- how long should I run my simulation ?
- should I eliminate the beginning of the simulation in order to wait until the system stabilizes ?
- I simulate a random way point model but the average speed in my simulation is not as expected. What has happened?
- What are confidence intervals ? How do I get them?

The goal of this course is to address these and other questions, in short: to be able to evaluate the performance of computer and communication systems, master the theoretical foundations of performance evaluation and the corresponding software packages.

Content:

Methodology A Performance Evaluation checklist. The scientific method. Dijkstra and Occam's principle. A Bit of **Simulation, Theory and Practice.** What happens in a simulator. Stationarity and ergodicity. Time and event averages and how they relate. Palm Calculus. Simulate arbitrary distributions. Perfect Simulation. **A Bit of Statistics.** Confidence intervals, exact and asymptotic methods. Tests. Factorial analysis. **Queuing Theory for Those Who Can't Wait.** Operational laws. Little and forced flows. Bottleneck analysis. Transients. Non-linearities. **Performance Phenomena.** Patterns and paradoxes. **Load and system models.** Self-similarity. Load forecasting. The Box-Jenkins method **Practicals** Using a statistics package (S-PLUS or Matlab). Measurements. Benchmarking. Load generation. SURGE. Discrete event simulation with ns2.

Required prior knowledge:

Cours de base en probabilité

Form of teaching:

Courses, laboratory, exercises, homework

URLs	1) http://ica1www.epfl.ch/perfeval				
Matière examinée / subjects examined	Performance evaluation				
Session	ETE	Coefficient / Crédits ECTS	7	Forme de l'examen / Form of examination	Oral

Titre	Périphériques
Title	Storage and display peripherals

Enseignant(s) / Instructor(s)	Gerlach Sebastian: IN, Hersch Roger: IN	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Informatique (2005-2006, Master semestre 1)	Cours: 2 Projet: 1		opt
Informatique (2005-2006, Master semestre 3)	Cours: 2 Projet: 1		opt

Objectifs:

Ce cours présente les méthodes et technologies utilisées pour interfacier des périphériques au niveau matériel, ainsi que pour accéder à ceux-ci au niveau logiciel. Le fonctionnement de divers types de périphériques est abordé, ainsi que les interfaces logicielles de bas et haut niveau sous Windows. Des mini-projets individuels permettent d'approfondir un sujet particulier.

Objectives:

This course presents methods and technologies for interfacing peripherals at the hardware level, and for accessing these at the software level. The inner workings of several types of peripherals are discussed, as well as the low and high level software interfaces in the Windows operating system. Mini-projects allow in-depth study of a particular subject.

Contenu:

Suite à la diversification des périphériques disponibles pour les ordinateurs individuels contemporains, il est important d'être capable de concevoir des systèmes utilisant ceux-ci de façon efficace. Interfaçage: Survol de la connectique actuelle: USB, Firewire, Bluetooth, PCI, AGP, PCI Express, principes de haut niveau, interface logicielle, plug and play. Stockage d'information: supports magnétiques, organisation des données sur disque, modélisation des disques, interfaces (SCSI, ATA, Serial ATA), disques magnéto-optiques, CD-ROM, DVD, archivage (bandes magnétiques), tableaux de disques RAID. Affichage: écrans graphiques, gestion de l'écran sous Windows (fenêtres), affichage d'images et de texte, impression. Interaction homme-machine: interfaces d'entrée-sortie (souris, joystick, clavier), interface USB, interface logicielle DirectPlay. Multimédia: caméras, microphones, haut-parleurs, flux continu, interface logicielle DirectShow. Les laboratoires et mini-projets offrent aux étudiants la possibilité d'implémenter les concepts présentés (gestion de blocs sur disque, conception de systèmes de fichiers, affichage, protocoles HID sur USB, etc.).

Content:

Due to the huge diversity of peripherals available for modern personal computers, it is critical to be able to develop systems that use them efficiently. Interfaces: Overview of current interfaces: USB, Firewire, Bluetooth, PCI, AGP, PCI Express, high level principles, software interfaces, plug and play. Storage: magnetic media, organisation of data on disks, performance modelling of disks, interfaces (SCSI, ATA, Serial ATA), magneto-optical disks, CD-ROM, DVD, archiving (magnetic tapes), RAID disk arrays. Display: graphic displays, display management in Windows (windowing), displaying graphics, images and text, printing. Human-computer interaction: input-output interfaces (mouse, joystick, keyboard), USB interface, DirectPlay software interface. Multimedia: cameras, microphones, loudspeakers, continuous streams, DirectShow software interface. The labs and mini-projects allow students to implement the concepts presented in the course (block-based disk access, file system design, display, HID protocols on USB, etc.)

Forme d'enseignement:

Cours, laboratoires (Windows avec Visual C++)

Form of teaching:

Course, laboratories (Windows avec Visual C++)

Bibliographie:

Cours polycopié et notes de laboratoire

Matière examinée / subjects examined		Périphériques			
Session	PRI	Coefficient / Crédits ECTS	3	Forme de l'examen / Form of examination	Oral

Titre	Projet d'informatique été
Title	Projet d'informatique été

Enseignant(s) / Instructor(s)	Profs divers *:		Langue / Language
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Informatique (2005-2006, Master semestre 2)	Projet: 12		

Objectifs:

Former les étudiants à la résolution de problèmes informatiques de manière autonome et présenter les résultats de leur recherche sous forme de mémoire et de défense orale.

Contenu:

Travaux de recherche individuelle à effectuer pendant le semestre, selon directives d'un professeur. Sujet du travail à choisir parmi la liste des sujets de travail de semestre accessible en permanence sur internet depuis l'adresse :

<http://sin.epfl.ch/>

Pour les étudiants intéressés à avoir une collaboration multidisciplinaire et intéressés aux aspects commerciaux, le projet I ou II EPFL peut être couplé avec un projet "business" fait par un étudiant HEC. Une séance d'information sera faite au début du semestre. Pour plus d'information, contactez le professeur Alain Wegmann (alain.wegmann@epfl.ch) pour les étudiants EPFL et Yves Pigneur (yves.pigneur@hec.unil.ch) pour les étudiants HEC !

Objectives:

To form students to resolve on their own computerscience problems. Presentation of the results of their research in a report and oral examination.

Content:

Individual research works to perform in the semester under the conduct of a C.S. professor. The subject will be chosen among the themes proposed by the Computer Science Department permanently accessible on the web from :

<http://sin.epfl.ch/>

For the students interested in multi-disciplinary collaboration and business issues, the EPFL project can be linked to a "business" project done by an HEC student. An information session will be organized at the beginning of the semester. For more information, you can contact the professor Alain Wegmann (alain.wegmann@epfl.ch) for the EPFL students and Yves Pigneur (yves.pigneur@hec.unil.ch) for the HEC students !

Matière examinée / subjects examined		Projet d'informatique été			
Session	ETE	Coefficient / Crédits ECTS	12	Forme de l'examen / Form of examination	Ctrl continu

Titre	Projet d'informatique hiver
Title	Projet d'informatique hiver

Enseignant(s) / Instructor(s)	Profs divers *:	Langue / Language	
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Informatique (2005-2006, Master semestre 1)	Projet: 12		
Informatique (2005-2006, Master semestre 3)	Projet: 12		

Objectifs:

Former les étudiants à la résolution de problèmes informatiques de manière autonome et présenter les résultats de leur recherche sous forme de mémoire et de défense orale.

Contenu:

Travaux de recherche individuelle à effectuer pendant le semestre, selon directives d'un professeur. Sujet du travail à choisir parmi la liste des sujets de travail de semestre accessible en permanence sur internet depuis l'adresse :

<http://sin.epfl.ch/>

Pour les étudiants intéressés à avoir une collaboration multidisciplinaire et intéressés aux aspects commerciaux, le projet EPFL peut être couplé avec un projet "business" fait par un étudiant HEC. Une séance d'information sera faite au début du semestre. Pour plus d'information, contactez le professeur Alain Wegmann (alain.wegmann@epfl.ch) pour les étudiants EPFL et Yves Pigneur (yves.pigneur@hec.unil.ch) pour les étudiants HEC !

Objectives:

To form students to resolve on their own computerscience problems. Presentation of the results of their research in a report and oral examination.

Content:

Individual research works to perform in the semester under the conduct of a C.S. professor. The subject will be chosen among the themes proposed by the Computer Science Department permanently accessible on the web from :

<http://sin.epfl.ch/>

For the students interested in multi-disciplinary collaboration and business issues, the EPFL project can be linked to a "business" project done by an HEC student. An information session will be organized at the beginning of the semester. For more information, you can contact the professor Alain Wegmann (alain.wegmann@epfl.ch) for the EPFL students and Yves Pigneur (yves.pigneur@hec.unil.ch) for the HEC students !

Matière examinée / subjects examined		Projet d'informatique hiver			
Session	PRI	Coefficient / Crédits ECTS	12	Forme de l'examen / Form of examination	Ctrl continu

Titre	Projet STS (été)
Title	STS project

Enseignant(s) / Instructor(s)	Galland Blaise: STS	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Informatique (2005-2006, Master semestre 2)	Projet: 4		opt
SC - EPFL (IS) (2005-2006, Master semestre 2)	Projet: 4		opt
SC - EPFL (IS) (2005-2006, Master semestre 4)	Projet: 4		opt

Objectifs:

Le but du projet STS est de mener une petite recherche sur les interactions entre la Science, la Technique et la Société. A travers ce travail, l'étudiant doit montrer qu'il maîtrise les principales variables de l'environnement qui déterminent l'appropriation sociale des technologies par ses usagers finaux : économiques, idéologiques, sociologiques, représentationnelles, éthiques, etc. Il lui est demandé de définir une problématique Science-Technique-Société, et de mettre en œuvre les moyens méthodologiques pour y donner une réponse dans un mémoire de 20 à 30 pages qu'il fera seul ou à deux.

Une variante plus commerciale est également proposée: le but est de faire un plan stratégique et un plan commercial pour une nouvelle entreprise. Pour les étudiants intéressés par une activité pluridisciplinaire, il est également possible de coupler cette variante avec un projet "business" fait par un étudiant HEC. Pour plus d'information, contactez le professeur Alain Wegmann (alain.wegmann@epfl.ch) pour les étudiants EPFL et Yves Pigneur (yves.pigneur@hec.unil.ch) pour les étudiants HEC.

Objectives:

The goal is to make a personal study investigating the interaction Science Technology and Society (STS). The goal of the project is to investigate an STS theme picked by the student or a group of students. The student must exhibit an understanding of the main factors that determine social benefit from technologies by end-users. He is supposed to identify a problematic situation as to the interaction between Science, Technology and Society, and to provide an answer in a 20-30 pages report using the adequate methods and tools for investigation.

A business-oriented option is also available, its goal is to write a strategic plan and a business plan for a new enterprise (using the concepts taught in the courses STS I to III). This project is run by groups of 2-3 people and is supervised by Prof. Wegmann. For the students interested in multi-disciplinary collaboration and business issues, this option can be tied to a "business" project done by an HEC student. For more information, you can contact Professor Alain Wegmann (alain.wegmann@epfl.ch) for the EPFL students and Yves Pigneur (yves.pigneur@hec.unil.ch) for the HEC students !

Matière examinée / subjects examined		Projet STS (été)			
Session	ETE	Coefficient / Crédits ECTS	6	Forme de l'examen / Form of examination	Ctrl continu

Titre	Projet STS (hiver)
Title	STS project

Enseignant(s) / Instructor(s)	Coray Giovanni: IN	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Informatique (2005-2006, Master semestre 1)	Projet: 4		opt
Informatique (2005-2006, Master semestre 3)	Projet: 4		opt
SC - EPFL (IS) (2005-2006, Master semestre 1)	Projet: 4		opt
SC - EPFL (IS) (2005-2006, Master semestre 3)	Projet: 4		opt

Objectifs:

Le but du projet STS est de mener une petite recherche sur les interactions entre la Science, la Technique et la Société. A travers ce travail, l'étudiant doit montrer qu'il maîtrise les principales variables de l'environnement qui déterminent l'appropriation sociale des technologies par ses usagers finaux : économiques, idéologiques, sociologiques, représentationnelles, éthiques, etc. Il lui est demandé de définir une problématique Science-Technique-Société, et de mettre en œuvre les moyens méthodologiques pour y donner une réponse dans un mémoire de 20 à 30 pages qu'il fera seul ou à deux.

Une variante plus commerciale est également proposée: le but est de faire un plan stratégique et un plan commercial pour une nouvelle entreprise.

Pour les étudiants intéressés par une activité pluridisciplinaire, il est également possible de coupler cette variante avec un projet "business" fait par un étudiant HEC. Pour plus d'information, contactez le professeur Alain Wegmann (alain.wegmann@epfl.ch) pour les étudiants EPFL et Yves Pigneur (yves.pigneur@hec.unil.ch) pour les étudiants HEC.

Objectives:

The goal is to make a personal study investigating the interaction Science Technology and Society (STS).

The goal of the project is to investigate an STS theme picked by the student or a group of students. The student must exhibit an understanding of the main factors that determine social benefit from technologies by end-users. He is supposed to identify a problematic situation as to the interaction between Science, Technology and Society, and to provide an answer in a 20-30 pages report using the adequate methods and tools for investigation.

A business-oriented option is also available & its goal is to write a strategic plan and a business plan for a new enterprise (using the concepts taught in the courses STS I to III). This project is run by groups of 2-3 people and is supervised by Prof. Wegmann.

For the students interested in multi-disciplinary collaboration and business issues, this option can be tied to a "business" project done by an HEC student. For more information, you can contact Professor Alain Wegmann (alain.wegmann@epfl.ch) for the EPFL students and Yves Pigneur (yves.pigneur@hec.unil.ch) for the HEC students !

URLs	1) http://ic.epfl.ch/page57512-fr.html				
Matière examinée / subjects examined	Projet STS (hiver)				
Session	PRI	Coefficient / Crédits ECTS	6	Forme de l'examen / Form of examination	Ctrl continu

Titre	Real-time embedded systems
Title	Real-time embedded systems

Enseignant(s) / Instructor(s)	Beuchat René: IN		Langue / Language	EN
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours	
Informatique (2005-2006, Master semestre 1)	Cours: 2 Projet: 2	6	opt	
Informatique (2005-2006, Master semestre 3)	Cours: 2 Projet: 2	6	opt	
Génie électrique et électronique (2005-2006, Master semestre 1)	Cours: 2 Projet: 2			
SC - EPFL (IS) (2005-2006, Master semestre 1)	Cours: 2 Projet: 2	6	opt	
SC - EPFL (IS) (2005-2006, Master semestre 3)	Cours: 2 Projet: 2	6	opt	

Objectifs:

Ce cours est orienté compréhension des microcontrôleurs spécialisés et utilisation de leurs interfaces programmables. Le lien important qui est à la base des systèmes embarqués entre le matériel, le langage assembleur et un langage de haut niveau (C) est mis en évidence. Les modèles de diverses familles de contrôleurs sont étudiés et mis en oeuvre dans des laboratoires pratiques. Les problèmes de la programmation temps réel sont mis en évidence dans une application de robot mobile qui est le fil conducteur du cours. La gestion des interruptions, de leur temps de réponse sont mis en évidence. Les outils de développement croisés sont utilisés.

Contenu:

Les thèmes principaux sont le traitement des interfaces A/D, D/A, timer, co-processeurs dédiés, interfaces séries, contrôles de moteurs et capteurs divers. Chaque thème est traité par un cours théorique et un laboratoire associé. L'ensemble des laboratoires est effectué sur des cartes microcontrôleur spécialement développées pour ce cours. Un robot mobile est entièrement programmé depuis les interfaces matérielles jusqu'à une application de contrôle du robot. Un système d'exploitation temps réel est étudié et utilisé avec les laboratoires.

Prérequis:

Introduction aux systèmes informatiques, Electronique, Systèmes logiques, Architecture des ordinateurs, Programmation (C/C++)

Forme d'enseignement:

Ex cathedra et exercices

Objectives:

This course is oriented on the teaching of specialized microcontroller and their programmable interfaces. The important link between hardware, assembly language, high level language (as C/C++) is show. Models of microcontroller family is studied and used in practical laboratories. Problems of real time programming are practically demonstrated with mobile robot experimentations. Interruptions, latency times, answer response time are put in evidence. Some cross developing tools are used.

Content:

A/D, D/A, timer, dedicated coprocessor, serial interfaces, motor controller and some captors interfaces are hardware main topics. Different processors as 68HC12, Psoc, ARM, NIOS are used in this courses, as well as embedded digital camera, for practical laboratories. A small mobile robot named Cyclope is used as material support for the specialized processor boards. The robot programming is done from the hardware interface to the real time application. A real time operating system is studied and used in the laboratories.

Required prior knowledge:

Introduction aux systèmes informatiques, Electronique, Systèmes logiques, Architecture des ordinateurs, Programmation (C/C++)

Form of teaching:

Ex cathedra and exercises

Matière examinée / subjects examined		Real-time embedded systems			
Session	PRI	Coefficient / Crédits ECTS	4	Forme de l'examen / Form of examination	Ctrl continu

Titre	Real-time programming
Title	Real-time programming

Enseignant(s) / Instructor(s)	Decotignie Jean-Dominique: SC		Langue / Language	EN
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours	
Informatique (2005-2006, Master semestre 1)	Cours: 3 Projet: 1	6	opt	
Informatique (2005-2006, Master semestre 3)	Cours: 3 Projet: 1	6	opt	
Génie électrique et électronique (2005-2006, Master semestre 1)	Cours: 3 Projet: 1	1	opt	
Systèmes de communication (2005-2006, Bachelor semestre 5)	Cours: 3 Projet: 1		opt	

Objectifs:

A l'issue du cours, l'étudiant aura acquis les connaissances principales liées à la conception et la réalisation des systèmes temps réel. Les différentes notions seront illustrées par des exercices et des laboratoires.

Contenu:

1. Introduction sur l'informatique du temps-réel et ses particularités
2. Modélisation des systèmes temps-réel - contexte, types
3. Modélisation asynchrone du comportement logique - Réseaux de Petri
4. Modélisation synchrone - GRAFCET (liens avec les langages synchrones)
5. Programmation des systèmes temps-réels - types de programmation (polling, par interruption, par états, exécutifs cycliques, coroutines, tâches)
6. Noyaux et systèmes d'exploitation temps-réel - problèmes, principes, mécanismes (tâches synchrones et asynchrones, synchronisation des tâches, gestion du temps et des événements)
7. Ordonnement - problèmes, contraintes, nomenclature
8. Ordonnement à priorités statiques (Rate Monotonic) et selon les échéances (EDF)
9. Ordonnement en tenant compte des ressources, des relations de précedence et des surcharges
10. Ordonnement de tâches multimedia
11. Evaluation des temps d'exécution
12. Introduction aux systèmes répartis temps-réel - définition, types de coopération, synchronisation d'horloges, communications, tolérance aux fautes.

Forme d'enseignement:

Ex cathedra, exercices et pratique

Bibliographie:

Polycopié

Objectives:

At the completion of the course, the student will have mastered the main topics concerning the design and programming of real-time systems. The course topics will be illustrated through exercises and a practical case study.

Content:

1. Introduction - Real-time systems and their characteristics
2. Modeling real-time systems - context and types
3. Asynchronous models of logical behavior - Petri nets
4. Synchronous models - GRAFCET (link with synchronous languages)
5. Programming real-time systems (polling, cyclic executives, co-routines, state based programming)
6. Real-time kernels and operating systems - problems, principles, mechanisms (synchronous and sporadic tasks, synchronization, event and time management)
7. Scheduling ζ problem, constraints, taxonomy
8. Fixed priority and deadline oriented scheduling
9. Scheduling in presence of shared resources, precedence constraints and overloads
10. Scheduling of continuous media tasks
11. Evaluation of worst case execution times
12. Introduction to distributed real-time systems

Form of teaching:

Ex cathedra, exercises and practical work

Matière examinée / subjects examined		Real-time programming			
Session	PRI	Coefficient / Crédits ECTS	4	Forme de l'examen / Form of examination	Ecrit

Titre	Real-time systems
Title	Real-time systems

Enseignant(s) / Instructor(s)	Decotignie Jean-Dominique: SC		Langue / Language	EN
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours	
Informatique (2005-2006, Master semestre 2)	Cours: 2	4	opt	
SC - EPFL (IS) (2005-2006, Master semestre 2)	Cours: 2	3	opt	
SC - EPFL (IS) (2005-2006, Master semestre 4)	Cours: 2	3	opt	

Objectifs:

A l'issue du cours, l'étudiant aura acquis les connaissances principales liées à la problématique et aux solutions apportées pour les communications temps réel dans les systèmes de transport et en contrôle de processus industriels. des systèmes temps réel. L'application de ces techniques au multimedia sera aussi abordée.

Contenu:

1. Introduction (Hiérarchie des communications, motivation pour les réseaux, types d'applications)
2. Besoins (délai, gigue, prévisibilité, topologie, coût, etc.)
3. Architecture des systèmes de communication et son influence sur le comportement temporel (modèle OSI, modèles d'interaction, approches architecturales - activation par événements ou temps, interconnexion)
4. Impact de la couche physique (topologie, cuivre, fibre, radio, sécurité intrinsèque, connecteurs)
5. Contrôle de l'accès au milieu et procédures de lien (trafic synchrone et asynchrone)
6. Les autres couches (réseau, transport, application, synchronisation d'horloge, gestion de réseau)
7. Détermination des garanties temporelles (ordonnancement, avec ou sans erreur)
8. Les bus de terrain. Analyse des solutions principales et de la satisfaction des besoins (Profibus, FIB, MVB, CAN, Asi, etc.)
9. Ethernet et le temps réel - problèmes et solutions
10. Les solutions sans fil (802.11, ZigBee, Bluetooth)

Forme d'enseignement:

Ex cathedra

Bibliographie:

Transparents et photocopiés

Objectives:

At the completion of the course, the student will have mastered the main problems and solutions related to communications under real-time constraints in transportation systems and in the control of industrial processes. Applications to multimedia will also be sketched.

Content:

1. Introduction (hierarchy in communications, motivation for networks, types of applications)
2. Requirements (delay, jitter, predictability, topology, cost, etc.)
3. Communication systems architecture and its influence on temporal behavior (OSI model, communication models, real-time paradigms: Time-Triggered vs. Event-Triggered, interworking)
4. Physical layer impact (topology, fibers, copper, wireless, intrinsic safety, connectors)
5. Medium Access Control and Logical Link Control (synchronous and asynchronous traffic)
6. Other layers (network, transport, application, clock synchronization, network management)
7. Real-Time performance assessment (scheduling, without error, in presence of errors)
8. Fieldbuses and analysis of the main solutions (Profibus, FIP, MVB, CAN, ASI, etc.) and how they fulfill the requirements
9. Ethernet and the many ways to offer real-time performances
10. Wireless solutions (802.11, Zigbee, Bluetooth)

Form of teaching:

Ex cathedra

Matière examinée / subjects examined		Real-time systems			
Session	ETE	Coefficient / Crédits ECTS	3	Forme de l'examen / Form of examination	Ecrit

Titre	Swarm intelligence
Title	Swarm intelligence

Enseignant(s) / Instructor(s)	Martinoli Alcherio: SC		Langue / Language	EN
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours	
Informatique (2005-2006, Master semestre 1)	Cours: 2 Projet: 3	1 6	opt	
Informatique (2005-2006, Master semestre 3)	Cours: 2 Projet: 3	1 6	opt	
SC - EPFL (IS) (2005-2006, Master semestre 1)	Cours: 2 Projet: 3	4 6	opt	
SC - EPFL (IS) (2005-2006, Master semestre 3)	Cours: 2 Projet: 3	4 6	opt	

Objectifs:

L'intelligence collective (IC) montrée par des sociétés animales telles que des colonies de fourmis ou des bancs de poissons a inspiré la création d'un nouveau paradigme de calcul et de comportement. Le but de ce cours est d'expliquer les mécanismes du comportement collectif de ces sociétés à travers des modèles mathématiques et de montrer comment ils peuvent être adaptés pour développer, par exemple, des algorithmes d'optimisation combinatoire innovateurs ou des architectures de contrôle distribuées pour des robots. Le cours est un mélange équilibré de théorie, de simulation, et d'expériences avec des outils matériels réels.

Contenu:

1. Introduction aux concepts de base tels que l'auto-organisation et la stigmergie ainsi qu'aux outils logiciels et matériels utilisés dans le cours.
2. Mouvements collectifs dans les sociétés animales et humaines ; mécanismes de récolte, suivi et création de piste, division du travail, agrégation et ségrégation, auto-assemblage et transport coopératif chez les insectes sociaux.
3. Méthodes de modélisation multi-niveau : simulation réalistes, modèle microscopiques et macroscopiques.
4. Algorithmes d'optimisation combinatoire (ACO, PSO) basés sur l'IC; comparaison avec des autres techniques d'optimisation multi-agents; algorithmes de classification de données et partition de graphes basés sur l'IC.
5. Applications de l'IC dans les télécommunications, la recherche opérationnelle, robotique collective, réseaux de capteurs et actuateurs.

Prérequis:

Cours de base en analyse, calcul de probabilité et programmation (C/C++ et Matlab)

Forme d'enseignement:

Ex cathedra et laboratoires assistés

Forme du contrôle:

Avec contrôle continu

Bibliographie:

Bonabeau, Dorigo, Theraulaz., Swarm Intelligence: From Natural to Artificial Systems, Oxford University Press, 1999. Articles spécifiques distribués à chaque leçon.

Objectives:

Swarm Intelligence (SI) is a new computational and behavioral paradigm for solving distributed problems; it is based on the principles underlying the behavior of natural systems consisting of many individuals, such as ant colonies and flocks of birds.

The student will be able to understand the underlying principles of collective behavior in natural systems through mathematical models and study their applications in engineering, from combinatorial optimization algorithms to swarm robotics.

The course is a well-balanced mixture of theory, simulation, and laboratory exercises using real hardware platforms.

Content:

1. Introduction to key concepts (e.g., self-organization, stigmergy) and software and hardware tools used in the course
2. Collective movements in animal and human societies; foraging, trail-laying and following, division of labor, aggregation and segregation, self-assembling, and collaborative transportation in social insects.
3. Multi-level modeling methodologies: realistic simulation, microscopic and macroscopic modeling.
4. SI-based combinatorial optimization (ACO, PSO); comparison with other multi-agent machine-learning techniques; data clustering and graph partitioning algorithms based on SI.
5. Applications of SI in telecommunication, operational research, collective robotics, and sensor and actuators networks.

Required prior knowledge:

Cours de base en analyse, calcul de probabilité et programmation (C/C++ et Matlab)

Form of teaching:

Ex cathedra and aided laboratory work

Form of examination:

With continuous control

URLs		1) http://swis.epfl.ch/teaching/SC741/			
Matière examinée / subjects examined		Swarm intelligence			
Session	PRI	Coefficient / Crédits ECTS	6	Forme de l'examen / Form of examination	Oral

Titre	Traitement automatique de la parole
Title	Automatic speech processing

Enseignant(s) / Instructor(s)	Bourlard Hervé: EL	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Informatique (2005-2006, Master semestre 1)	Cours: 2 Exercice: 1		opt
Informatique (2005-2006, Master semestre 3)	Cours: 2 Exercice: 1		opt
Systèmes de communication (2005-2006, Bachelor semestre 5)	Cours: 2 Exercice: 1		opt

Objectifs:

L'objectif de ce cours est de présenter les principaux formalismes, modèles et algorithmes permettant la réalisation d'applications mettant en oeuvre des techniques de traitement de la parole (codage, analyse/synthèse, reconnaissance)

Contenu:

1. Introduction: Tâches du traitement de la parole, domaines d'applications de l'ingénierie linguistique.
2. Outils de base: Analyse et propriétés spectrales du signal de parole, reconnaissance statistique de formes (statiques), programmation dynamique.
3. Codage de la parole: Propriétés perceptuelles de l'oreille, théorie de la quantification, codage dans le domaine temporel et fréquentiel.
4. Synthèse de la parole: Analyse morpho-syntaxique, transcription phonétique, prosodie, modèles de synthèse.
5. Reconnaissance de la parole: Classification de séquences et algorithme de déformation temporelle dynamique (DTW), systèmes de reconnaissance à base de chaînes de Markov cachées (HMM).
6. Reconnaissance et vérification du locuteur : Formalisme, test d'hypothèse, HMM pour la vérification du locuteur.
7. Ingénierie linguistique: état de l'art et applications types.

Forme d'enseignement:

Ex cathedra

Bibliographie:

Traitement de la parole, PPUR

Objectives:

The goal of this course is to provide the students with the main formalisms, models and algorithms required for the implementation of advanced speech processing applications (involving, among others, speech coding, speech analysis/synthesis, and speech recognition).

Content:

1. Introduction: Speech processing tasks, language engineering applications.
2. Basic Tools: Analysis and spectral properties of the speech signal, linear prediction algorithms, statistical pattern recognition, programming dynamic.
3. Speech Coding: Human hearing properties, quantization theory, speech coding in the temporal and frequency domains
4. Speech Synthesis: morpho-syntactic analysis, phonetic transcription, prosody, speech synthesis models.
5. Automatic speech recognition: Temporal pattern matching and Dynamic Time Warping (DTW) algorithms, speech recognition systems based on Hidden Markov Models (HMM).
6. Speaker recognition and speaker verification: Formalism, hypothesis testing, HMM based speaker verification.
7. Linguistic Engineering: state-of-the-art and typical applications

Matière examinée / subjects examined		Traitement automatique de la parole			
Session	PRI	Coefficient / Crédits ECTS	3	Forme de l'examen / Form of examination	Ecrit

Titre	Virtual reality
Title	Virtual reality

Enseignant(s) / Instructor(s)	Thalmann Daniel: IN	Langue / Language	EN
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Informatique (2005-2006, Master semestre 2)	Cours: 2 Exercice: 1	3	opt

Objectifs:

Ce cours présente les concepts et les méthodes pour réaliser des environnements virtuels. pouvant être distribués sur les réseaux multimédias. On introduit ainsi des concepts avancés pour l'interaction 3D, la reconnaissance de gestes, les interfaces haptiques, le son spatial, la communication faciale, la reconnaissance et la synthèse de la parole. On montre comment créer des avatars et des populations autonomes dans les mondes virtuels. On insiste sur des applications concrètes comme les téléconférences 3D, la téléchirurgie ou les systèmes de simulation en cas d'urgence interactive.

Contenu:

1. INTRODUCTION. Concepts de base des environnements virtuels, matériel, logiciel, applications
2. INTERACTION MULTIMODALE. capture de mouvements, reconnaissance de gestes, reconnaissance et synthèse de la parole, son spatial, interfaces haptiques
3. ENVIRONNEMENTS VIRTUELS DANS LA COMMUNICATION MULTIMEDIA . Environnements virtuels distribués, avatars, communication faciale
4. VIE ARTIFICIELLE DANS LES ENVIRONNEMENTS VIRTUELS. Sens virtuels, perception-action, créatures autonomes
5. REALITE AUGMENTEE. Mélange réel-virtuel, « tracking », calibration de caméras
6. APPLICATIONS. Téléconférences 3D, téléchirurgie, jeux vidéo 3D, systèmes de simulation

Prérequis:

Computer graphics

Forme d'enseignement:

Ex cathedra, vidéo, exerc. sur station graphique

Forme du contrôle:

Avec contrôle continu

Bibliographie:

Notes de cours

Objectives:

This course presents the concepts and methods to define complex virtual environments, which may be distributed on multimedia networks. We introduce advanced concepts for 3D interaction, gesture recognition, haptic interfaces, spatial sound, facial communication, speech recognition and synthesis. We show how to create avatars or 3D clones, how to create autonomous people in virtual worlds. We emphasize concrete applications like 3D teleconferences, tele-surgery or systems for emergency and training..

Content:

1. INTRODUCTION. Basic concepts of virtual environments, hardware, software, applications
2. MULTIMODAL INTERACTION. motion capture, gesture recognition, speech recognition and synthesis, spatial sound, haptics
3. VIRTUAL ENVIRONNEMENTS IN THE MULTIMEDIA COMMUNICATION. Distributed Virtual Environments, avatars, facial communication
4. ARTIFICIAL LIFE IN VIRTUAL ENVIRONNEMENTS. Virtual sensors, perception-action, autonomous
5. AUGMENTED REALITY. Mixed reality, tracking, camera calibration
6. APPLICATIONS. 3D teleconferences, tele-surgery, 3D video-games, training systems

Required prior knowledge:

Computer graphics

Form of teaching:

Ex cathedra, video, exercises on graphic workstation

Form of examination:

With continuous control

Matière examinée / subjects examined		Virtual reality			
Session	ETE	Coefficient / Crédits ECTS	4	Forme de l'examen / Form of examination	Ecrit



ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

SECTION D'INFORMATIQUE

Options de spécialisations

Hors plan d'Etudes

2005 / 2006

Titre	Analyse de données génétiques
Title	Statistical analysis of genetic data

Enseignant(s) / Instructor(s)	Morgenthaler Stephan: MA	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Mathématiques (2005-2006, Master semestre 1)	Cours: 2 Exercice: 2	2	opt
SC - EPFL (IS) (2005-2006, Master semestre 1)	Cours: 2 Exercice: 2	4	opt
SC - EPFL (IS) (2005-2006, Master semestre 3)	Cours: 2 Exercice: 2	4	opt

Objectifs:

Présenter les idées de base de la modélisation statistique des processus de procréation, de mutation, de sélection et d'évolution. Etudier les modèles biomathématiques de carcinogénèse et présenter quelques idées concernant la génétique moléculaire.

Contenu:

- Carcinogénèse, modèles à multiple frappes, modèles à deux étapes
- Gènes et génotypes, ségrégation mendélienne, fréquences d'allèles et équilibre de Hardy-Weinberg
- Risque génétique pour des maladies
- Phénotypes, estimation de fréquences d'allèles, algorithme EM
- Chromosomes, liaison génétique, déséquilibre
- Mutations, polymorphismes, marqueurs génétiques, l'effet d'une taille restreinte d'une population
- Sélection
- Propagation d'un caractère : composantes de variation, hérédité
- Données moléculaires : alignement de séquences et recherche de prototypes
- Modèles pour l'évolution des espèces
- Reconstruction de phylogénies: méthodes basées sur des matrices de distances, méthodes de vraisemblance.

Prérequis:

Notions élémentaires de probabilités et statistique

Forme d'enseignement:

Cours ex cathedra et exercices en classe

Forme du contrôle:

Contrôle des exercices

Bibliographie:

Liste de livres distribuées pendant le cours

Objectives:

This course offers the students an introduction to the field of statistical genomics in the form of models for procreation, for genetic variability and mutations, for natural selection and for evolution. Biomathematical models for carcinogenesis will be discussed and some basic ideas in the area of computational molecular biology will be given.

Content:

- Carcinogenesis, multi-hit models, two-stage models
- Genes and genotypes, Mendelian segregation, allele frequencies and Hardy-Weinberg equilibrium
- Genetic risk in diseases
- Phenotypes, estimation of allele frequencies, EM algorithm
- Chromosomes, genetic linkage, disequilibrium
- Mutations, polymorphisms, genetic markers, effects of finite population size
- Selection
- Inheritance of quantitative traits: components of variance, heritability
- Molecular data: sequence alignment, sequence patterns
- Models for the evolution of species
- Phylogeny construction: distance matrix methods, likelihood methods.

Required prior knowledge:

Basic notions in probability and statistics

Form of teaching:

Ex cathedra lecture and classroom exercises

Form of examination:

Exercises checking

Matière examinée / subjects examined		Analyse de données génétiques			
Session	PRI	Coefficient / Crédits ECTS	4	Forme de l'examen / Form of examination	Oral

Titre	Automatique I
Title	Control systems I

Enseignant(s) / Instructor(s)	Longchamp Roland: GM	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Génie électrique et électronique (2005-2006, Bachelor semestre 5)	Cours: 2 Exercice: 1		
Génie mécanique (2005-2006, Bachelor semestre 5)	Cours: 2 Exercice: 1		
Microtechnique (2005-2006, Bachelor semestre 5)	Cours: 2 Exercice: 1		

Objectifs:

L'étudiant maîtrisera les méthodes classiques d'analyse et de synthèse des régulateurs automatiques. Il sera en outre capable de modéliser les systèmes discrets en vue de leur commande par ordinateur. L'étudiant sera en mesure d'analyser et de synthétiser des régulateurs numériques.

La partie pratique porte sur l'étude expérimentale du comportement de systèmes dynamiques et de certains concepts de base introduits aux cours d'Automatique I et II, ainsi que la mise en oeuvre de systèmes de mesure et de commande.

Contenu:

- Introduction à l'automatique
- Commande par ordinateur de processus
- Echantillonnage et reconstruction
- Systèmes discrets
- Transformée en z
- Fonction de transfert discrète du système bouclé
- Réponse harmonique

Prérequis:

Analyse complexe, Systèmes dynamiques

Préparation pour:

Automatique II
Identification et commande I, II
Systèmes multivariables I, II

Forme d'enseignement:

Ex cathedra. Démonstrations et exercices intégrés.

Forme du contrôle:

Examen écrit au printemps

Bibliographie:

R. Longchamp, Commande numérique de systèmes dynamiques, PPUR, 2005

Objectives:

I + II The student will know how to analyze and design classical control systems. Moreover, he will be able to model discrete-time systems for the purpose of digital control, and will be able to analyze and design digital control systems.

The practical activities are dedicated to the experimental study of dynamic systems and some basic control concepts introduced in the Control Systems course, as well as the implementation of measurement and control solutions.

Content:

- Introduction to control systems
- Digital control systems
- Sampling and reconstruction
- Discrete-time systems
- The z-transform
- Closed-loop discrete-time transfer function
- Frequency response

Matière examinée / subjects examined		Automatique I			
Session	PRI	Coefficient / Crédits ECTS	3	Forme de l'examen / Form of examination	Écrit

Titre	Automatique II
Title	Control systems II and Laboratory Work

Enseignant(s) / Instructor(s)	Gillet Denis: GM, Longchamp Roland: GM	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Génie mécanique (2005-2006, Bachelor semestre 6)	Cours: 2 TP: 1 TP: 1		opt
Microtechnique (2005-2006, Bachelor semestre 6)	Cours: 2 TP: 1 TP: 1		

Objectifs:

L'étudiant maîtrisera les méthodes classiques d'analyse et de synthèse des régulateurs automatiques. Il sera en outre capable de modéliser les systèmes discrets en vue de leur commande par ordinateur. L'étudiant sera en mesure d'analyser et de synthétiser des régulateurs numériques.

La partie pratique porte sur l'étude expérimentale du comportement de systèmes dynamiques et de certains concepts de base introduits aux cours d'Automatique I et II, ainsi que la mise en oeuvre de systèmes de mesure et de commande.

Contenu:

- Stabilité
- Numérisation
- Synthèse directe

Prérequis:

Automatique I

Préparation pour:

Identification et commande I, II
Systèmes multivariables I, II

Forme d'enseignement:

Ex cathedra. Démonstrations et exercices intégrés.

Forme du contrôle:

Cours à option, examen écrit en été

Bibliographie:

R. Longchamp, Commande numérique de systèmes dynamiques, PPUR, 2005

Objectives:

The student will know how to analyze and design classical control systems. Moreover, he will be able to model discrete-time systems for the purpose of digital control, and will be able to analyze and design digital control systems.

The practical activities are dedicated to the experimental study of dynamic systems and some basic control concepts introduced in the Control Systems course, as well as the implementation of measurement and control solutions.

Content:

- Stability
- Translation of analog design
- Discrete-time design

Matière examinée / subjects examined		Automatique II			
Session	ETE	Coefficient / Crédits ECTS	3	Forme de l'examen / Form of examination	Ecrit

Titre	Color imaging
Title	Color imaging

Enseignant(s) / Instructor(s)	Süsstrunk Sabine: SC	Langue / Language	EN
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
SC - EPFL (IS) (2005-2006, Master semestre 1)	Cours: 2 Exercice: 1	2	opt
SC - EPFL (IS) (2005-2006, Master semestre 3)	Cours: 2 Exercice: 1	2	opt

Objectifs:

Les systèmes de traitement d'images négligent souvent les non-linéarités du système visuel humain, et la couleur est simplement traitée comme une extension tridimensionnelle d'un système à valeurs de gris. De ce fait, nombre d'algorithmes ne sont pas efficaces lorsqu'ils essaient de tenir compte de la couleur - par exemple lors de recherches automatiques d'images dans une base de données - et les images résultant de ces algorithmes sont souvent de qualité insuffisante. Ce cours apprend aux étudiants à appliquer les modèles du système visuel humain pour résoudre des problèmes de codage et de traitement des images de couleur. Les étudiants devront aussi réaliser un mini-projet basé dans l'un des domaines d'application.

Contenu:

1. Révision de la vision en couleurs et de la colorimétrie
2. Modèles de couleurs et différences de couleurs
3. Physique de la formation des images en couleurs
4. Codage des couleurs et espaces de couleurs
5. Systèmes d'images de couleur
6. Systèmes d'images multispectrales
7. Détection de la lumière ambiante et invariance des couleurs
8. Adaptation chromatique
9. Modèles de perception des couleurs
10. Comment reproduire des couleurs qui semblent "naturelles"
11. Affichage des images et théorie rétinex
12. Critères perceptifs de qualité des images
13. Applications: gestion des couleurs, compression d'images en couleurs, segmentation basée sur la couleur, Caractérisation et calibrage des appareils, reconstructions d'images, archivage d'images, etc.

Préparation pour:

Color reproduction

Forme d'enseignement:

Ex cathedra et exercices

Bibliographie:

Notes de cours polycopiées, articles

Objectives:

The non-linear processing of the human visual system is often neglected in the development of imaging systems, and color is regarded as « only » a three-dimensional extension to image processing algorithms. As a result, many algorithms are not as efficient for complex tasks, such as automatic image retrieval, and/or the resulting image quality not sufficient for many applications. This course teaches students to apply the knowledge of how the human visual system processes color information to solve color image encoding and processing tasks. The students will also implement a mini-project based on one of the application topics.

Content:

1. Review of color vision and colorimetry.
2. Color models and color difference formulae.
3. Physics of color image formation.
4. Color encoding and color spaces.
5. Color imaging systems
6. Multispectral imaging systems.
7. Illuminant detection and color constancy.
8. Chromatic adaptation.
9. Color appearance models.
10. Naturalness of color image reproduction.
11. Image rendering and retinex
12. Image quality and psychophysics.
13. Applications : color management, color image compression, color segmentation, device calibration and characterization, image reconstruction, image archiving, etc.

Prerequisite for:

Color reproduction

Form of teaching:

Ex cathedra and exercises

Matière examinée / subjects examined		Color imaging			
Session	PRI	Coefficient / Crédits ECTS	4	Forme de l'examen / Form of examination	Oral

Titre	E-Business
Title	E-Business

Enseignant(s) / Instructor(s)	Pigneur Yves: SC		Langue / Language	EN
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours	
SC - EPFL (IS) (2005-2006, Master semestre 1)	Cours: 4 Exercice: 2	5	opt	
SC - EPFL (IS) (2005-2006, Master semestre 3)	Cours: 4 Exercice: 2	5	opt	

Objectifs:

Le phénomène Internet a certes connu quelques vicissitudes, mais le commerce électronique et le e-business ont redéfini les façons de conduire les affaires, en fournissant notamment des nouveaux modèles d'affaires. Ce cours doit fournir aux participants une compréhension en profondeur du e-business. Le cours présente les concepts nécessaires pour appréhender et modéliser les business models ou modèles d'affaire, les processus d'affaire et les services e-business. Le cours est construit à partir d'études de cas, de lectures, de présentations de groupe, et de recherches personnelles

Contenu:

A la fin du cours, les participants

- Auront une bonne compréhension du e-business,
- Seront sensibilisés à l'alignement entre les affaires et les solutions informatiques
- Seront familiers avec les recherches récentes en e-business et systèmes d'information

Plus spécifiquement, les étudiants seront capables de

- Décrire le modèle d'affaire d'une entreprise,
- Analyser et résumer quelques théories en vigueur dans ce domaine, et
- Appliquer ces théories à la conception de processus d'affaire et de services e-business.

Forme d'enseignement:

Interactif, études de cas, devoirs

Forme du contrôle:

Participation, projet, test

Bibliographie:

Liste écrite

Objectives:

Despite the dot-com bubble burst, e-business and e-commerce have redefined the ways of conducting business, providing new business models, and competing in the global marketplace. This course provides the participants with an understanding of e-business and e-commerce from a business perspective, in a Internet-enabled economy. The course introduces concepts, frameworks, and models for defining business models, designing inter-organization business processes, describing e-commerce services, and assessing technology environments. Learning will be accomplished through lectures, case studies, group presentation, and research.

Content:

Upon completion this course, the participants will

- have a general understanding of the current state and trends of e-business and e-commerce,
- be aware of the desirable business/IT alignment,
- be familiar with the state of the art of the research in e-business and e-commerce.

More specifically, the participants will be able to

- design the business model of a company,
- analyze and summarize theories current in e-business, and
- apply these theories for designing a e-business process or a e-business service.

Form of teaching:

Interactive, case studies, assignment

Form of examination:

Participation, project, test

URLs	1) http://www.hec.unil.ch/yp/GTI/				
Matière examinée / subjects examined	E-Business				
Session	PRI	Coefficient / Crédits ECTS	6	Forme de l'examen / Form of examination	Ecrit

Titre	Gestion de production I
Title	Production management I

Enseignant(s) / Instructor(s)	Gardon Rémy: GM	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Génie mécanique (2005-2006, Master semestre 1)	Cours: 2	2 3	opt
Microtechnique (2005-2006, Master semestre 1)	Cours: 2	3	opt
Programme Management de la technologie et entrepreneuriat (2005-2006, Master semestre 1)	Cours: 2		opt
Programme Management de la technologie et entrepreneuriat (2005-2006, Mineur semestre 1)	Cours: 2		opt

Objectifs:

L'étudiant doit être capable de

1. Comprendre les aspects principaux du fonctionnement de l'entreprise de production en tant que système et reconnaître les principaux types d'organisations de la production.
2. Maîtriser les bases des éléments fondamentaux de la production et de la logistique interne (nomenclatures, gestion des besoins, gestion des stocks, méthodes de planification, de suivi et d'ordonnement)
3. Comprendre le fonctionnement et les critères d'optimisation de la gestion de stock. Connaître les méthodes de réapprovisionnement et dimensionner les paramètres de gestion sur une base statistique.
4. Maîtriser les principes de fonctionnement de la planification de production sur une base MRP. Comprendre et appliquer les méthodes de planification des ressources.

Contenu:

- l'entreprise de production en tant que système ; les flux de matière, d'information et financier ; les défis technico-économiques ; les types d'organisations de production
- la structure des coûts et des produits, nomenclatures et codification
- la gestion des stocks ; méthodes de réapprovisionnement, dimensionnement statistique des niveaux de gestion, bases d'optimisation, mesure des performances
- planification et suivi de la production ; niveaux de gestion, plan industriel et commercial, méthodes MRP, plan directeur de production.

Prérequis:

cours obligatoires du plan d'étude

Préparation pour:

cours Gestion de production II, projets de semestre et de Master

Forme d'enseignement:

cours ex cathedra, travaux de groupe, présentation d'étudiants, exercices et lectures individuelles hors cours, études de cas.

Forme du contrôle:

Contrôle continu et examen oral

Bibliographie:

notes polycopiées et livres de références

Objectives:

The student should be capable of

1. Understanding the main characteristics of the manufacturing enterprise as a system and the major types of production organizations.
2. Mastering the basic elements of the production and internal logistic (bill of material, demand and inventory management, planning, control and scheduling)
3. Understanding the working principles and the optimization criteria of inventory management. Using the replenishment methods and calculating the parameters on a statistical basis.
4. Mastering the working principles of production planning on an MRP basis. Understanding and applying the capacity planning methods.

Content:

- the manufacturing enterprise as a system; material, information and financial flows; the various production organization types
- the product and cost structures; bill of material and codification
- inventory management; replenishment methods, statistical determination of the control levels, optimization and performance criteria.
- production planning and control; levels of planning, general industrial plan, the MRP method, master production scheduling plan.

Matière examinée / subjects examined		Gestion de production I,II			
Session	ETE	Coefficient / Crédits ECTS	4	Forme de l'examen / Form of examination	Oral

Titre	Gestion de production II
Title	Production management II

Enseignant(s) / Instructor(s)	Gardon Rémy: GM		Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours	
Génie mécanique (2005-2006, Master semestre 2)	Cours: 2	2 3	opt	
Microtechnique (2005-2006, Master semestre 2)	Cours: 2	3	opt	
Programme Management de la technologie et entrepreneuriat (2005-2006, Master semestre 2)	Cours: 2		opt	
Programme Management de la technologie et entrepreneuriat (2005-2006, Mineur semestre 2)	Cours: 2		opt	

Objectifs:

L'étudiant doit être capable de :

1. Comprendre le fonctionnement de la génération des besoins, ses enjeux et ses limites. Choisir et appliquer les méthodes mathématiques de prévision.
2. Comprendre les principes et les limites des méthodes de gestion des flux basées sur les principes du juste à temps. Dimensionner des systèmes KANBAN.
3. Comprendre les nouveaux défis et les développements récents en gestion de production et logistique interne. Identifier les avantages, inconvénients, limites et contraintes de méthodes mixtes de gestion de la production.

Contenu:

- la génération des besoins, objectifs, moyens, contraintes ; types de prévisions, méthodes mathématiques et subjectives de prévision; méthodes mixtes.
- le juste à temps, objectifs, principes de base; la méthode KANBAN, dimensionnement des systèmes KANBAN, heuristiques; conditions de fonctionnement et limites des méthodes JIT.
- évolution de la gestion de production, les nouveaux défis; méthodes mixtes de gestion de la production; méthode OPT, DSSPL; nouveaux développements et perspectives.

Prérequis:

Gestion de production I

Préparation pour:

projets de semestre et diplôme

Forme d'enseignement:

cours ex cathedra, travaux de groupes, présentation d'étudiants, exercices et lectures individuelles hors cours, études de cas.

Forme du contrôle:

Contrôle continu, étude de cas et examen oral

Bibliographie:

notes polycopiées et livres de références

Objectives:

1. Understanding the working principles of the demand determination, its challenges, constraint and limitations. Choosing and applying the mathematical forecasting methods.
2. Understanding the characteristics and limitations of production planning and control methods based on the just in time principle. Designing and dimensioning KANBAN systems.
3. Understanding the new challenges of and the most recent developments in production planning and control and in inbound logistic. Identifying the advantages, disadvantages, limitations and constrains of mixed production planning methods.

Content:

- demand management, goals, methods, constraint; types of forecasts, mathematical and subjective forecasting methods; mixed methods.
- just in time; objectives, basic principles; the KANBAN method, dimensioning of KANBAN systems, heuristics; functioning conditions and limitations of JIT methods.
- evolution of production planning and control; the new challenges; mixed methods in production planning and control; methods OPT and DSSPL; new developments and future trends

Matière examinée / subjects examined		Gestion de production I,II			
Session	ETE	Coefficient / Crédits ECTS	4	Forme de l'examen / Form of examination	Oral

Titre	Hardware systems modeling I
Title	Hardware systems modeling I

Enseignant(s) / Instructor(s)	Vachoux Alain: EL	Langue / Language	EN
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Génie électrique et électronique (2005-2006, Master semestre 1)	Cours: 2	1	obl
Microtechnique (2005-2006, Master semestre 1)	Cours: 2	2	opt
SC - EPFL (IS) (2005-2006, Master semestre 1)	Cours: 2	6	opt
SC - EPFL (IS) (2005-2006, Master semestre 3)	Cours: 2	6	opt

Objectifs:

- Etre capable de créer des modèles VHDL de composants numériques pour la simulation et la synthèse.
- Etre capable de créer des modèles de test et d'appliquer des techniques de vérification.
- Acquérir des règles de modélisation.
- Disposer d'une bibliothèque de modèles VHDL.
- Obtenir une connaissance pratique des outils de simulation et de synthèse VHDL.
- Situer VHDL par rapport à d'autres langages (Verilog, SystemC)

Contenu:

Introduction

Notion de modèle et de langages de description de matériel. Principes de la simulation logique et de la synthèse logique et architecturale. Caractéristiques de VHDL (langage, flot de conception, règles de modélisation).

VHDL pour la synthèse

Sous-ensemble synthétisable standard du langage (IEEE Std 1076.3 et 1076.6). Synthèse d'instructions VHDL.

Modélisation de composants numériques

Eléments combinatoires et séquentiels. Contrôleurs (machines à états finis). Unités arithmétiques (additionneurs, multiplieurs, ALU). Mémoires (registres, RAM, ROM, FIFO, LIFO). Filtrés numériques. Circuits d'interface (UART, PCI), Processeurs. Modèles de test et techniques de vérification.

VHDL vs. Verilog et SystemC

Caractéristiques des langages Verilog et SystemC avec exemples. Comparaison avec VHDL. Techniques de modélisation communes.

Prérequis:

Outils informatiques (module VHDL); Systèmes logiques

Préparation pour:

Hardware systems modeling II; VLSI design II

Forme d'enseignement:

Cours avec exemples et exercices pratiques intégrés

Bibliographie:

Notes polycopiées, précis de syntaxe VHDL

Objectives:

- To be able to create VHDL models of digital components for simulation and synthesis.
- To be able to create testbench models and to learn verification techniques.
- To learn modeling guidelines.
- To develop a reference library of VHDL models.
- To get a working knowledge of VHDL simulation and synthesis tools.
- To position VHDL with respect to other hardware description languages (Verilog, SystemC).

Content:

Introduction

Models in electronic design automation. Hardware description languages. Logic simulation. Architectural and logic synthesis. VHDL characteristics (language, design flow, modeling guidelines).

Synthesis with VHDL

VHDL synthesis subset (IEEE Std 1076.3 and 1076.6). Synthesis of VHDL statements.

Modelling of digital components

Basic combinational and sequential elements. Controllers (finite state machines). Arithmetic units (adders, multipliers, ALU). Memories (registers, RAM, ROM, FIFO, LIFO). Digital filters. Interface circuits (UART, PCI). Processors. Testbenches and verification techniques.

VHDL vs. Verilog and SystemC

Verilog and SystemC characteristics with examples. Comparison with VHDL. Common modeling techniques.

Matière examinée / subjects examined		Hardware systems modeling I			
Session	PRI	Coefficient / Crédits ECTS	2	Forme de l'examen / Form of examination	Ecrit

Titre	Hardware systems modeling II
Title	Hardware systems modeling II

Enseignant(s) / Instructor(s)	Vachoux Alain: EL	Langue / Language	EN
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Génie électrique et électronique (2005-2006, Master semestre 2)	Cours: 2	1	obl
Microtechnique (2005-2006, Master semestre 2)	Cours: 2	2	opt
SC - EPFL (IS) (2005-2006, Master semestre 2)	Cours: 2	6	opt
SC - EPFL (IS) (2005-2006, Master semestre 4)	Cours: 2	6	opt

Objectifs:

- Etre capable de créer des modèles VHDL-AMS de composants analogiques et mixtes pour la simulation.
- Etre capable de créer des modèles de test et d'appliquer des techniques de vérification.
- Acquérir des règles de modélisation.
- Disposer d'une bibliothèque de modèles VHDL-AMS.
- Obtenir une connaissance pratique d'un outil de simulation VHDL-AMS.
- Situer VHDL-AMS par rapport à d'autres langages (Verilog-AMS, SystemC-AMS)

Contenu:

Introduction

Notion de modèle et de langages de description de matériel.
Techniques de la simulation analogique et mixte.

Le langage VHDL-AMS

Caractéristiques de VHDL-AMS (langage, flot de conception, règles de modélisation). Organisation d'un modèle VHDL-AMS. Modélisation comportementale et structurelle analogique et mixte.

Modélisation de composants analogiques

Primitives électriques. Amplificateur opérationnel, OTA. Filtrés. PLL. Modèles de test et techniques de vérification.

Modélisation de composants mixtes

Interfaces A/N et N/A. Convertisseurs A/N et N/A. PLL. CDR. Modèles de test et techniques de vérification.

VHDL-AMS vs. Verilog-AMS et SystemC-AMS

Caractéristiques des langages Verilog-AMS et SystemC-AMS. Comparaison avec VHDL-AMS.

Prérequis:

Outils informatiques (module VHDL)
Hardware systems modeling I

Forme d'enseignement:

Cours avec exemples et exercices pratiques intégrés

Bibliographie:

Notes polycopiées, précis de syntaxe VHDL-AMS

Objectives:

- To be able to create VHDL-AMS models of analog and mixed-signal components for simulation.
- To be able to create testbench models and to use verification techniques.
- To learn modeling guidelines.
- To develop a reference library of VHDL-AMS models.
- To get a working knowledge of a VHDL-AMS simulation tool.
- To position VHDL-AMS with respect to other hardware description languages (Verilog-AMS, SystemC-AMS).

Content:

Introduction

Models in electronic design automation. Hardware description languages. Analog and mixed-signal simulation techniques.

The VHDL-AMS language

VHDL-AMS characteristics (language, design flow, modelling guidelines). VHDL-AMS model organization. Behavioural and structural VHDL-AMS modeling.

Modelling of analog components

Electrical primitives. Operational amplifier, OTA. Filters. PLL. Testbenches and verification techniques.

Modelling of mixed-signal components

A/D and D/A interfaces. A/D and D/A converters. PLL. CDR. Testbenches and verification techniques.

VHDL-AMS vs. Verilog-AMS and SystemC-AMS

Verilog-AMS and SystemC-AMS characteristics with examples. Comparison with VHDL-AMS.

Matière examinée / subjects examined		Hardware systems modeling II			
Session	ETE	Coefficient / Crédits ECTS	2	Forme de l'examen / Form of examination	Ecrit

Titre	Identification et commande I
Title	Identification and control I

Enseignant(s) / Instructor(s)	Karimi Alireza: GM	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Génie électrique et électronique (2005-2006, Master semestre 1)	Cours: 2	3	opt
Génie mécanique (2005-2006, Master semestre 1)	Cours: 2	2 3 4 5	opt
Microtechnique (2005-2006, Master semestre 1)	Cours: 2	3 4	opt

Objectifs:

L'étudiant apprendra à modéliser des systèmes dynamiques sur la base de mesures entrée-sortie. Il maîtrisera les possibilités offertes par certains logiciels modernes d'analyse et d'identification (Identification Toolbox de Matlab).

Contenu:

- Types de modèles dynamiques
- Méthode de corrélation
- Analyse spectrale
- Modèles paramétriques
- Identification des paramètres
- Validation du modèle
- Aspects pratiques de l'identification
- Projet en groupe

Prérequis:

Automatique I et II

Préparation pour:

Identification et commande II

Forme d'enseignement:

Cours avec exemples, exercices et projet individuel

Forme du contrôle:

oral

Bibliographie:

Cours polycopié "Identification de systèmes dynamiques"

Objectives:

This course covers the identification of dynamic systems, i.e. the modeling of these systems on the basis of input/output data. The possibilities offered by modern software packages such as the Identification Toolbox of Matlab for both system identification and control system analysis will be discussed.

Content:

- Model types
- Correlation method
- Spectral analysis
- Parametric models
- Parameter identification
- Model validation
- Practical aspects of identification
- Group project

Matière examinée / subjects examined		Identification et commande I			
Session	PRI	Coefficient / Crédits ECTS	2	Forme de l'examen / Form of examination	Oral

Titre	Identification et commande II
Title	Identification and control II

Enseignant(s) / Instructor(s)	Karimi Alireza: GM, Longchamp Roland: GM	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Génie électrique et électronique (2005-2006, Master semestre 2)	Cours: 2	3	opt
Génie mécanique (2005-2006, Master semestre 2)	Cours: 2	2 3 4 5	opt
Microtechnique (2005-2006, Master semestre 2)	Cours: 2	3 4	opt

Objectifs:

L'étudiant sera en mesure de synthétiser des régulateurs polynomiiaux. Il pourra réaliser des régulateurs adaptatifs et maîtrisera des algorithmes d'auto-ajustement des régulateurs PID.

Contenu:

- Régulateur RST polynomial
- Commande adaptative
- Auto-ajustement des régulateurs PID

Prérequis:

Automatique I, II, Identification et commande I

Forme d'enseignement:

Ex cathedra. Démonstrations et exercices intégrés.

Forme du contrôle:

oral

Bibliographie:

R. Longchamp, Commande numérique de systèmes dynamiques, PPUR, 2005.

Objectives:

The student will be able to design polynomial controllers. Moreover, he will know to implement adaptive controllers and how to automatically tune PID controllers.

Content:

- RST polynomial controller
- Adaptive control
- Auto-tuning of PID controllers

Matière examinée / subjects examined		Identification et commande II			
Session	ETE	Coefficient / Crédits ECTS	2	Forme de l'examen / Form of examination	Oral

Titre	Infochimie
Title	Infochemistry

Enseignant(s) / Instructor(s)	Röthlisberger Ursula: CGC, Tavernelli Ivano: CGC	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Chimie et génie chimique (2005-2006, Bachelor semestre 6)	Cours: 1 Projet: 3		opt

Objectifs:

Introduction à la théorie et les applications pratiques des méthodes de la structure électronique et des techniques de la modélisation moléculaire.

Contenu:

Répétition brève des concepts fondamentaux de la mécanique quantique et des algorithmes numériques utilisés pour les implémentations pratiques. Principes essentiels des méthodes de la structure électronique : HF, MPn, CI, CC, DFT. Résumé des techniques computationnelles pour la modélisation des systèmes moléculaires.

Prérequis:

Algèbre linéaire, Mécanique quantique

Forme d'enseignement:

ex cathedra et projet par ordinateur

Remarque:

Enseignement partiel du module Cheminformatique

Bibliographie:

"Quantum Chemistry", A. Szabo; "Molecular Modelling", A.R. Leach

Objectives:

Introduction to the theory and practical application of quantum chemical electronic structure methods and molecular modelling techniques.

Content:

Short repetition of the basic concepts of quantum mechanics and the main numerical algorithms used for practical implementations. Basic principles of electronic structure methods: Hartree-Fock, many-body perturbation theory, configuration interaction, coupled-cluster theory, density functional theory. Overview of computational molecular modelling techniques.

Application of these techniques in a practical research project.

Matière examinée / subjects examined		Dynamique moléculaire et simulation Monte-Carlo et Eléments de bioinformatique et Infochimie			
Session	ETE	Coefficient / Crédits ECTS	9	Forme de l'examen / Form of examination	Ctrl continu

Titre	Mathematical modelling of DNA I
Title	Mathematical modelling of DNA I

Enseignant(s) / Instructor(s)	Maddocks John: MA	Langue / Language	EN
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Mathématiques (2005-2006, Master semestre 2)	Cours: 2 Exercice: 2	2	opt
SC - EPFL (IS) (2005-2006, Master semestre 2)	Cours: 2 Exercice: 2	4	opt
SC - EPFL (IS) (2005-2006, Master semestre 4)	Cours: 2 Exercice: 2	4	opt

Objectifs:

Ce cours de deux semestres vise à introduire, dans le contexte particulier de l'ADN, les interactions entre analyse, simulation numérique et résultats expérimentaux, interactions qui constituent l'essence de la modélisation mathématique.
En plus des étudiants intéressés à la modélisation de l'ADN, ce cours se destinera aussi à ceux qui désirent une introduction générale au processus de modélisation mathématique, et couvrira diverses techniques mathématiques et numériques couramment rencontrées dans ce domaine.

Contenu:

(voir aussi : http://lcvmwww.epfl.ch/dna_main.html)

1. INTRODUCTION

- La molécule d'ADN (Structure, Fonction)
- Motivations expérimentales pour la modélisation

2. MODELES ET TYPES D'ANALYSES

- Modèles (Modèles discrets, Modèle élastique continu)
- Analyse (Statique, Dynamique, Statistique)

3. EQUILIBRES DES MODELES CONTINUS DE TIGES

- Théorie élémentaire des tiges
- Connexion entre les paramètres et l'ADN
- Equations de l'équilibre (conditions de bord en 2 points)
- Techniques mathématiques
- Calcul des variations
- Formulation Hamiltonienne
- Théorie de bifurcation et rôle des symétries
- Stabilité des équilibres
- Simulations numériques : Discrétisation spatiale
- Continuation de paramètres
- Exemple: Circularisation de l'ADN.

Prérequis:

Premier cycle en math. ou physique, (ou avec permission de l'enseignant)

Forme d'enseignement:

Cours ex cathedra, avec exercices en classe

Forme du contrôle:

Examen oral

Bibliographie:

Distribuée au début du cours

Objectives:

This two part course is designed to be an introduction, within the particular context of DNA, to the interplay between analysis, computation and experiment that makes up the process called mathematical modelling.

In addition to students mainly interested in DNA modelling, the course is intended for students wishing an introduction to the modelling process in general, and will describe a number of widely encountered mathematical and computational techniques.

Content:

(see also : http://lcvmwww.epfl.ch/dna_main.html)

1. INTRODUCTION

- The DNA molecule (Structure, Function)
- Experimental motivations for modelling

2. DNA MODELS AND TYPES OF ANALYSES

- Models (Discrete models, Continuum elastic rod model)
- Analysis (Statics, Dynamics, Statistics)

3. EQUILIBRIUM PROBLEMS IN CONTINUUM ROD MODELS

- Basic rod theory
- Connection of parameters to DNA experiments
- Equilibrium equations (2 point boundary-value problem)
- Mathematical techniques
- Calculus of variations
- Hamiltonian formulation
- Bifurcation theory and role of symmetries
- Stability of equilibria
- Numerical computation
- Space discretization
- Parameter continuation
- Example: DNA Cyclization.

Form of teaching:

Ex cathedra lecture with exercises in the classroom

Form of examination:

Oral exam

Matière examinée / subjects examined		Mathematical modelling of DNA I			
Session	ETE	Coefficient / Crédits ECTS	4	Forme de l'examen / Form of examination	Oral

Titre	Mathematical modelling of DNA II
Title	Mathematical modelling of DNA II

Enseignant(s) / Instructor(s)	Maddocks John: MA	Langue / Language	EN
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Mathématiques (2005-2006, Master semestre 2)	Cours: 2 Exercice: 2	2	opt
SC - EPFL (IS) (2005-2006, Master semestre 2)	Cours: 2 Exercice: 2	4	opt
SC - EPFL (IS) (2005-2006, Master semestre 4)	Cours: 2 Exercice: 2	4	opt

Objectifs:

Contenu:

(voir aussi : http://lcvmwww.epfl.ch/dna_main.html)

Suite de "MODELISATION MATHEMATIQUE DE L'ADN (I)".

4. GEOMETRIE ET TOPOLOGIE DU TWIST ET DU VRILLAGE

- Nombre d'enlacement
- Twist
- Vrillage

5. MECANIQUE STATISTIQUE DES MODELES DE CHAINES

- Distribution de probabilité (i.e. Maxwell-Boltzman)
- Analyse de divers modèles de chaînes (y.c. les concepts de longueur de Kuhn et de longueur de persistance)
- Simulations numériques (méthode de Monte-Carlo)

6. LES PROBLEMES DYNAMIQUES DANS LES MODELES DISCRETS ET CONTINUS

- Le frottement et l'agitation thermique
- L'équation de Fokker-Planck - Dynamique de Langevin
- L'équation de Smoluchowski - Dynamique Brownienne
- Contraintes
- Coordonnées généralisées / Fonctions d'énergie non-séparable
- Décomposition en mode normaux
- Discrétisation temps/espace.

Prérequis:

Mathematical Modelling of DNA (I) (ou avec permission de l'enseignant)

Forme d'enseignement:

Cours ex cathedra, avec exercices en classe

Forme du contrôle:

Examen oral

Bibliographie:

Distribuée au début du cours.

Content:

(see also : http://lcvmwww.epfl.ch/dna_main.html)

Continuation of "MATHEMATICAL MODELLING OF DNA (I)".

4. GEOMETRY AND TOPOLOGY OF TWIST AND WRITHE

- Link
- Twist
- Writhe

5. STATISTICAL MECHANICS OF CHAIN MODELS

- Probability distribution (e.g. Maxwell-Boltzman)
- Analysis within various chain models (including concepts of Kuhn and persistence lengths)
- Numerical computation (Monte-Carlo)

6. DYNAMICAL PROBLEMS IN DISCRETE AND CONTINUUM MODELS

- Viscous and stochastic loads
- Fokker-Planck Equation - Langevin Dynamics
- Smoluchowski Equation - Brownian Dynamics
- Constraints
- General Coordinates/Non-separable Energy Functions
- Normal mode decomposition
- Time/Space discretization.

Required prior knowledge:

Mathematical Modelling of DNA (I) (or with the teacher's lecture)

Form of teaching:

Ex cathedra lecture with exercises in the classroom

Form of examination:

Oral exam

Matière examinée / subjects examined		Mathematical modelling of DNA II			
Session	ETE	Coefficient / Crédits ECTS	4	Forme de l'examen / Form of examination	Oral

Titre	Mécatronique
Title	Mechatronics

Enseignant(s) / Instructor(s)	Colombi Silvio: EL	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Génie électrique et électronique (2005-2006, Master semestre 2)	Cours: 2	3	opt

Objectifs:

La mécatronique est un domaine interdisciplinaire en pleine expansion se basant sur la **mécanique classique**, **l'électronique** et **l'informatique**.

L'objectif de cet enseignement est d'illustrer, par différents exemples d'applications réels, comment **améliorer une solution mécanique** en utilisant des actionneurs, des capteurs, de l'électronique et des algorithmes de réglage. Ces exemples d'applications montrent différents aspects de la conception mécatronique et sont une importante source d'inspiration pour beaucoup d'autres applications. L'étudiant sera sensibilisé au fait que la conception d'un système est toujours une "question de compromis".

Contenu:

Spécification et conception d'un système mécatronique

Conception mécatronique: coût, performances, approche système, diagramme d'influence, équivalents mécaniques, étapes de conceptions, outils de conception et de simulation, prototypage rapide : de la simulation à la réalité, méthodologie de conception.

Exemples d'applications choisis

Servomécanismes bilatéraux maître-esclave à retour de force, actionneurs et réglages pour un servomanipulateur maître-esclave à retour de force, "durcissement" électronique de transmission mécaniques, "durcissement" et linéarisation électronique d'actionneurs ; réglage du gros transporteur Boom de JET, compensation électronique des forces/couples parasites de moteurs synchrones à aimants permanents, compensation du frottement mécanique dans des applications "motion control", sustentation et guidage magnétique d'un véhicule, réglage d'un robot parallélogramme, suspension active d'une roue, dispositifs anti-blocage et anti-patinage, différentiel électronique, injecteur pour moteur à gaz naturel, réglage et commande d'un moteur à pistons libres.

Forme d'enseignement:

Ex cathedra

Forme du contrôle:

Oral

Bibliographie:

Cours et notes photocopiés

Objectives:

"Mechatronics" is a rapidly growing field, resulting from the combination of classical **electrical engineering**, **mechanical engineering** and **computer science**.

The goal of this teaching is to illustrate, through several real application examples, how **to improve a mechanical solution** using actuators, sensors, electronics and control algorithms. The examples show various features of the mechatronics design and are an important source of inspiration for many other applications. The student will be aware of the fact that a design is always a "question of compromise".

Content:

Specification and design of mechatronic systems

Design of mechatronic systems: cost, performances, system approach, diagram of influence, mechanical equivalents, design steps, simulation and design tools, rapid prototyping: from the simulation to the reality, design methodology.

Selected application examples

Bilateral Master-Slave force reflecting servomechanisms, Actuators and controls for a master-slave force reflecting servomanipulator, Electronic stiffening of mechanical transmissions, Electronic stiffening and linearisation of actuators; control of the JET Boom, Electronic compensation of the parasitic forces/torques of brushless DC motors, Friction compensation in motion control applications, Magnetic levitation and lateral guidance of a vehicle, Control of a parallelogram robot, Active suspension of a wheel, Anti-slip and anti-skid devices, Electronic differential, Injector for a natural gas engine, Command and control of a free pistons engine.

Matière examinée / subjects examined		Mécatronique			
Session	ETE	Coefficient / Crédits ECTS	2	Forme de l'examen / Form of examination	Oral

Titre	Modèles stochastiques pour les communications
Title	Stochastic models in communication

Enseignant(s) / Instructor(s)	Dousse Olivier: SC, Thiran Patrick: SC		Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours	
Systèmes de communication (2005-2006, Bachelor semestre 5)	Cours: 4 Exercice: 2			

Objectifs:

Maîtriser les outils des processus aléatoires utilisés par un ingénieur en systèmes de communication.

Contenu:

1. Rappels de probabilité: axiomes de probabilité, variable aléatoire et vecteur aléatoire.
2. Processus stochastiques à temps continu et à temps discret : analyse du second ordre (stationarité, ergodisme, densité spectrale, relations de Wiener- Khintchine, réponse d'un système linéaire invariant à des entrées aléatoires, processus gaussien, processus ARMA, filtres de Wiener).
3. Processus de Poisson et bruit impulsif de Poisson.
4. Chaînes de Markov à temps discret. Chaînes ergodiques, comportement asymptotique, chaînes absorbantes, temps d'attente, marches aléatoires simples, processus de branchement.
5. Chaînes de Markov à temps continu. Processus de naissance et de mort à l'état transitoire et stationnaire. Files d'attente simples: définition, loi de Little, files M/M/1... M/M/s/K, M/G/1.

Prérequis:

Cours de base en probabilité, analyse et algèbre linéaire

Préparation pour:

Cours en Systèmes de Communication à l'EPFL et à Eurécom

Forme d'enseignement:

Ex cathedra avec exercices

Forme du contrôle:

Avec contrôle continu

Objectives:

To acquire a working knowledge of the tools of random processes used by a communication systems engineer.

Content:

1. Review of probability: axioms of probability, random variable and random vector.
2. Continuous-time and discrete-time stochastic processes: second-order analysis (stationarity, ergodism, spectral density, Wiener-Khintchine relations, response of a LTI system to random inputs, Gaussian processes, ARMA processes, Wiener filter).
3. Poisson process and Poisson shot noise.
4. Discrete-time Markov chains. Ergodic chains, asymptotic behavior, absorbing chains, reaching time, simple random walks, branching processes.
5. Continuous-time Markov chains. Birth and death process: transient and steady-state analysis. Simple queues: definitions, Little's law, M/M/1... M/M/s/K, M/G/1 queues.

Form of teaching:

Ex cathedra with exercises

Form of examination:

With continuous control

Matière examinée / subjects examined		Modèles stochastiques pour les communications			
Session	PRI	Coefficient / Crédits ECTS	6	Forme de l'examen / Form of examination	Ecrit

Titre	Optimisation I
Title	Optimization I

Enseignant(s) / Instructor(s)	Bierlaire Michel: MA		Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours	
Génie électrique et électronique (2005-2006, Master semestre 1)	Cours: 2 Exercice: 1		opt	
Génie mécanique (2005-2006, Bachelor semestre 5)	Cours: 2 Exercice: 1		obl	
Systèmes de communication (2005-2006, Bachelor semestre 5)	Cours: 2 Exercice: 1		opt	

Objectifs:

Le cours a pour but d'initier les étudiants à la théorie de l'optimisation afin de leur permettre d'utiliser des algorithmes et des logiciels de manière adéquate, en appréciant leurs limitations méthodologiques et en interprétant correctement les résultats.

Objectives:

The course is an introduction to optimization theory, aimed at helping the students to appropriately use optimization algorithms and packages. The stress will be made on methodological issues and results analysis.

Contenu:

1. Introduction à l'optimisation
 - Modélisation, transformations du problème
2. Optimisation sans contrainte : analyse du problème
 - Définition du problème
 - Convexité / concavité ; Différentiabilité
 - Conditionnement et préconditionnement
3. Optimisation sans contrainte : conditions d'optimalité
4. Résolution de systèmes d'équations non linéaires
 - Méthode de Newton
 - Méthodes quasi-Newton
5. Optimisation sans contrainte : algorithmes
 - Problèmes quadratiques : gradients conjugués
 - Recherche linéaire
 - Région de confiance
 - Méthodes quasi-Newton
 - Problèmes de moindres carrés - Filtre de Kalman
6. Optimisation avec contraintes : analyse du problème
 - Contraintes actives
 - Qualification des contraintes
 - Elimination des contraintes
7. Introduction à la dualité.

Content:

1. Introduction to optimization
 - Modeling, problem transformations
2. Unconstrained optimization: problem analysis
 - Problem definition
 - Convexity / concavity; differentiability
 - Conditioning and preconditioning
3. Unconstrained optimization: optimality conditions
4. Solving systems of nonlinear equations
 - Newton's method
 - Quasi-Newton methods
5. Unconstrained optimization: algorithms
 - Quadratic problems: conjugate gradients
 - Linesearch
 - Trust region
 - Quasi-Newton methods
 - Least squares problems - Kalman filter
6. Constrained optimization : problem analysis
 - Active constraints
 - Constraints qualification
 - Constraints elimination
7. Introduction to duality.

Prérequis:

Algèbre linéaire, Analyse

Required prior knowledge:

Linear algebra, Analysis

Préparation pour:

Pratique des sciences de l'ingénieur

Prerequisite for:

Practice of engineering sciences

Forme d'enseignement:

Cours ex cathedra + travaux pratiques sur ordinateur

Form of teaching:

Ex cathedra lecture and practical works on computer

Bibliographie:

Bierlaire, M. Introduction à l'optimisation différentiable, PPUR (à paraître).
D. P. Bertsekas, Nonlinear programming, Athena Scientific, 1995.

Matière examinée / subjects examined		Optimisation I			
Session	PRI	Coefficient / Crédits ECTS	3	Forme de l'examen / Form of examination	Ecrit

Titre	Optimisation II
Title	Optimization II

Enseignant(s) / Instructor(s)	Prodon Alain: MA	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Systèmes de communication (2005-2006, Bachelor semestre 6)	Cours: 2 Exercice: 1		opt

Objectifs:

Se familiariser avec les méthodes de l'optimisation discrète. Connaître les fondements de méthodes efficaces spécifiques et de méthodes générales, leurs limitations, et les appliquer à des problèmes de configuration, routage, placement, ordonnancement.

Objectives:

To get acquainted with discrete optimization methods. To know the foundations of specific efficient methods, general methods, their limitations. Apply these to configuration, routing, placement and scheduling problems.

Contenu:

1. Introduction
 - Rappels de complexité
 - Modélisation et exemples
2. Optimisation combinatoire
 - Motivation et exemples
 - Graphes, chemins, circuits et problèmes de connectivité
 - Flot maximum
 - Flot à coût minimum
 - Multiflots
 - Affectations et couplages
3. Programmation entière et mixte
 - Motivation et exemples
 - Choix de formulations de problèmes
 - Méthodes de relaxations et Branch and Bound
 - Méthodes de plans coupants et Branch and Cut
 - Méthodes heuristiques, quêtes locales, recuit simulé, tabou, schémas d'approximation.

Content:

1. Introduction
 - Review of complexity
 - Modeling and examples
2. Combinatorial optimization
 - Motivation and examples
 - Graphs, paths, circuits and connectivity problems
 - Maximum flow
 - Minimum cost flow
 - Multiflow
 - Assignments and matchings
3. Integer and mixed integer programming
 - Motivation and examples
 - Choices in problem formulations
 - Relaxation methods and Branch and Bound
 - Cutting plane methods and Branch and Cut
 - Heuristic methods, local search, simulated annealing, tabu search, approximation schemes.

Prérequis:

Recherche opérationnelle, Algèbre linéaire

Required prior knowledge:

Operational research, linear algebra

Forme d'enseignement:

Cours ex cathedra, exercices en classe et sur ordinateur

Form of teaching:

Ex cathedra, exercises in the classroom and with computers

Bibliographie:

G. Nemhauser, L. Wolsey: Integer and Combinatorial Optimization, Wiley 1988.

URLs	1) http://				
Matière examinée / subjects examined	Optimisation II				
Session	ETE	Coefficient / Crédits ECTS	3	Forme de l'examen / Form of examination	Oral

Titre	Recherche opérationnelle
Title	Operations research

Enseignant(s) / Instructor(s)	Spada Michela: MA	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Systèmes de communication (2005-2006, Bachelor semestre 5)	Cours: 2 Exercice: 1		

Objectifs:

Les étudiants sont familiarisés avec :

- les principaux modèles de la recherche opérationnelle;
- la modélisation mathématique de processus techniques, logistiques et de gestion, en vue de l'optimisation des décisions sous-jacentes;
- l'utilisation de techniques d'optimisation, également en présence d'éléments stochastiques.

Contenu:

Programmation linéaire

Modélisation à l'aide de la programmation linéaire. Méthode du simplexe

Dualité, post-optimisation et méthode duale du simplexe.

Programmation paramétrique.

Systèmes d'inégalités linéaires, polyèdres, lemme de Farkas

Notions des ensembles et fonctions convexes

Problèmes d'optimisation associés

Optimisation séquentielle

Programmation dynamique déterministe

Applications : plus court chemin, problèmes de gestion des stocks, problème du sac à dos

Optimisation dans les graphes

Connexité, arbres, chaînes, chemins, cycles, circuits

Le problème du transbordement

Arbres couvrants de poids maximum

Applications à la modélisation

Prérequis:

Analyse, algèbre linéaire, informatique

Préparation pour:

Conception et gestion de systèmes de communication, algorithmique

Bibliographie:

Notes polycopiées

J.-F. Hêche, Th. M. Liebling, D. De Werra,

Recherche opérationnelle pour ingénieurs, tomes I et II

Objectives:

Students will be thoroughly familiar with :

- the various operations research models
- the mathematical modeling of processus, from technology, logistics and management, in due of optimizing the underlying decisions
- the use of optimization techniques also in a stochastic environment.

Content:

Linear programming

Formulating lp models. Simplex algorithm

Duality, post-optimization, dual simplex method. Parametric programming.

Linear inequality systems, polyhedra

Convex sets and functions

Associated optimization problems

Sequential optimization

Deterministic dynamic programming

Applications : shortest path problem, inventory problems, knapsack problem

Optimization problems in Graphs

Connexity, trees, chains, paths, cycle, circuits, description, matrices

Transshipment problem

Maximum weight spanning trees

Modeling applications

URLs	1) http://roso.epfl.ch/cours/optimisation_l/				
Matière examinée / subjects examined	Recherche opérationnelle				
Session	PRI	Coefficient / Crédits ECTS	3	Forme de l'examen / Form of examination	Ecrit

Titre	Réseaux de neurones et modélisation biologique
Title	Neural networks and biological modeling

Enseignant(s) / Instructor(s)	Gerstner Wulfram: IN	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Génie électrique et électronique (2005-2006, Master semestre 2)	Cours: 2 Exercice: 1		opt
Mathématiques (2005-2006, Master semestre 2)	Cours: 2 Exercice: 1	3	opt
Science et génie des matériaux (2005-2006, Master semestre 2)	Cours: 2 Exercice: 1	4	opt
Physique (2005-2006, Master semestre 2)	Cours: 2 Exercice: 1		opt
SC - EPFL (IS) (2005-2006, Master semestre 2)	Cours: 2 Exercice: 1	4	opt
SC - EPFL (IS) (2005-2006, Master semestre 4)	Cours: 2 Exercice: 1	4	opt

Objectifs:

Les réseaux de neurones sont une classe de modèles de traitement d'information inspirée par la biologie du cerveau. Ce domaine interdisciplinaire a attiré beaucoup d'intérêt parmi des mathématiciens, physiciens, informaticiens et biologistes. Le cours introduit les réseaux de neurones comme modèle du système nerveux. Il couvre la modélisation d'un neurone isolé, les groupes de neurones ainsi que les phénomènes d'apprentissage et d'adaptation.

Contenu:

1. Introduction (le cerveau comparé à l'ordinateur; les neurones; le problème de codage)
- I. Modèles de neurones isolés**
2. Modèles ioniques (modèle de Hodgkin et Huxley)
3. Modèles en 2 dimensions (modèle de Fitzhugh-Nagumo, analyse en espace de phase)
4. Modèles impulsionsnels d'un neurone (modèle "integrate-and-fire, spike response model")
5. Bruit et variabilité dans des modèles impulsionsnels (processus ponctuel, renewal process, résonance stochast.)
- II. Neurones connectés**
6. Groupes de neurones (activité d'une population, état asynchrone, oscillations)
7. Transmission des signaux par des populations (linéarisation de la dynamique, analyse signal et bruit)
8. Oscillations
9. Réseaux spatiaux continus
- III. Synapses et la base d'apprentissage**
10. La règle de Hebb (Long-term-potential et formul math.)
11. Analyse en composantes principales (apprentissage non-supervisé, règle de Oja)
12. Applications au système visuel et auditif (développement des champs récepteurs, localisation des sources sonores)
13. La mémoire associative (le modèle de Hopfield, relation au modèle de ferromagnétisme)

Prérequis:

Analyse I-III, Algèbre linéaire, Probabilité et statistique, Dynamical Systems Theory for Engineers

Forme d'enseignement:

Ex cathedra et exercices

Bibliographie:

Dayan & Abbott : Theoretical Neuroscience, MIT Press 2001; Gerstner & Kistler : Spiking Neuronmodels, Cambridge Univ. Press

Objectives:

Neural networks are a fascinating interdisciplinary field where physicists, biologists, and computer scientists work together in order to better understand the information processing in biology (visual system, auditory system, associative memory). In this course, mathematical models of biological neural networks are presented and analyzed

Content:

1. Introduction (brain vs computer; neurons and neuronal connections; the problem of neural coding)
- I. Models of single neurons**
2. Models on the level of ion current (Hodgkin-Huxley model)
3. Two-dimensional models and phase space analysis (Fitzhugh-Nagumo and Morris LeCar model)
4. Spiking neurons (integrate-and-fire and spike response model)
5. Noise and variability (point processes, renewal process, stochastic resonance)
- II. Networks**
6. Population dynamics (cortical organisation, population activity, asynchronous states)
7. Signal transmission by populations of neurons (linearized equations, signal transfer function)
8. Oscillations
9. Continuous field models
- III. Synapses and learning**
10. The Hebb rule and correlation based learning (long-term potentiation, spike-based and rate-based learning)
11. Principal Component Analysis (unsupervised learning, Oja's rule, normalization)
12. Applications: Visual and Auditory System (development of receptive fields, sound source localization)
13. Associative memory (Hopfield model; relation to ferromagnetic systems)

Required prior knowledge:

Analyse I-III, Algèbre linéaire, Probabilité et statistique, Dynamical Systems Theory for Engineers

Form of teaching:

Ex cathedra and exercises

URLs	1) http://diwww.epfl.ch/w3mantra/mantra_cours_ph.html				
Matière examinée / subjects examined	Réseaux de neurones et modélisation biologique				
Session	ETE	Coefficient / Crédits ECTS	3	Forme de l'examen / Form of examination	Ecrit

Titre	Systèmes multivariables I
Title	Multivariable systems I

Enseignant(s) / Instructor(s)	Müllhaupt Philippe: GM	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Génie électrique et électronique (2005-2006, Master semestre 1)	Cours: 2	3	obl
Génie mécanique (2005-2006, Master semestre 1)	Cours: 2	1 2 3 4 5	opt
Microtechnique (2005-2006, Master semestre 1)	Cours: 2	4	opt

Objectifs:

Ce cours traite de la conception de commandes numériques basée sur des méthodes d'état, ainsi que de la modélisation et de l'estimation d'état de systèmes dynamiques multivariables.

Contenu:

- Représentation par variables d'état de systèmes continus et discrets
- Conversion entre les représentations par fonction de transfert et par variables d'état
- Observabilité, gouvernabilité et stabilité
- Estimation d'état et observateur de Luenberger
- Contre-réaction d'état par placement de pôles
- Commande optimale quadratique (LQR)
- Commande prédictive

Prérequis:

Systèmes dynamiques, Automatique I et II

Préparation pour:

Systèmes multivariables II

Forme d'enseignement:

Cours ex cathedra avec exemples et exercices intégrés

Forme du contrôle:

oral

Bibliographie:

Cours polycopié "Systèmes multivariables I", Digital Control of Dynamic Systems, G.F. Franklin and al., Addison Wesley

Objectives:

This course covers the design of digital control systems using state-space methods, including the modeling and the state estimation of multivariable dynamic systems.

Content:

- State-variable representation of continuous and discrete systems
- State-space to/from transfer function conversion
- Observability, controllability and stability
- State estimation and Luenberger observer
- State feedback using pole placement
- Linear quadratic regulator (LQR)
- Predictive control

Matière examinée / subjects examined		Systèmes multivariables I			
Session	PRI	Coefficient / Crédits ECTS	2	Forme de l'examen / Form of examination	Oral

Titre	Systèmes multivariables II
Title	Multivariable systems II

Enseignant(s) / Instructor(s)	Müllhaupt Philippe: GM	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Génie électrique et électronique (2005-2006, Master semestre 2)	Cours: 2	3	obl
Génie mécanique (2005-2006, Master semestre 2)	Cours: 2	1 2 3 4 5	opt
Microtechnique (2005-2006, Master semestre 2)	Cours: 2	4	opt

Objectifs:

Ce cours introduit les méthodes de base d'analyse et de commande des systèmes non linéaires.

Contenu:

- Notions générales sur les systèmes non linéaires
- Description du comportement dans l'espace de phase
- Méthode de l'équivalent harmonique
- Analyse de stabilité par la méthode de Lyapunov
- Aperçu des stratégies de commande non linéaire

Prérequis:

Automatique I et II, Systèmes multivariables I

Forme d'enseignement:

Ex cathedra avec exemples et exercices intégrés.

Forme du contrôle:

oral

Bibliographie:

Notes de cours
Hassan K. Khalil « Nonlinear Systems », Prentice Hall 3rd edition, 2002

Objectives:

This course introduces the analysis and control methods for nonlinear systems.

Content:

- Nonlinear systems fundamentals
- Phase plane description of nonlinear dynamics
- Describing function analysis
- Lyapunov stability analysis
- Nonlinear control overview

Matière examinée / subjects examined		Systèmes multivariables II			
Session	ETE	Coefficient / Crédits ECTS	2	Forme de l'examen / Form of examination	Oral

Titre	Traitement d'images I
Title	Image processing I

Enseignant(s) / Instructor(s)	Unser Michaël: MT	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Microtechnique (2005-2006, Master semestre 1)	Cours: 3	1 2 3 4 5	opt
SC - EPFL (IS) (2005-2006, Master semestre 1)	Cours: 3	2 4	opt
SC - EPFL (IS) (2005-2006, Master semestre 3)	Cours: 3	2 4	opt

Objectifs:

Introduction aux techniques de base du traitement d'images. Initiation au développement en JAVA et à la mise en oeuvre d'algorithmes de traitement d'images; application à des exemples concrets en vision industrielle et en imagerie biomédicale.

Contenu:

- Introduction. Traitement et analyse d'images. Applications. Eléments d'un système de traitement.
- Caractérisation des images de type continu. Classe d'images. Transformée de Fourier 2D. Systèmes invariants par translation.
- Acquisition d'images. Théorie d'échantillonnage. Systèmes d'acquisition. Histogramme et statistiques simples. Quantification linéaire et Max-Lloyd.
- Caractérisation des images discrètes et filtrage linéaire. Transformée en z. Convolution. Séparabilité. Filtrage RIF et RII.
- Opérations de traitement d'images. Opérateurs ponctuels (seuillage, modification d'histogramme). Opérateurs spatiaux (lissage, rehaussement, filtrage non-linéaire). Opérateurs morphologiques simples.
- Introduction à l'analyse d'image et à la vision par ordinateur. Segmentation, détection de contours, détection d'objets, comparaison d'images

Prérequis:

Signaux et systèmes I, II

Préparation pour:

Traitement d'images II + projets

Forme d'enseignement:

Cours ex cathedra, exercices et travaux pratiques sur ordinateur

Bibliographie:

Notes polycopiées

Objectives:

Introduction to the basic techniques of image processing. Introduction to image processing software development and prototyping in JAVA; application to real-world examples in industrial vision and biomedical imaging.

Content:

- Introduction. Image processing versus image analysis. Applications. System components.
- Characterization of continuous images. Image classes. 2D Fourier transform. Shift-invariant systems.
- Image acquisition. Sampling theory. Acquisition systems. Histogram and simple statistics. Linear and Max-Lloyd Quantization.
- Characterization of discrete images and linear filtering. z-transform. Convolution. Separability. FIR and IIR filters.
- Image processing operations. Point operators (thresholding, histogram modification). Spatial operators (smoothing, enhancement, non-linear filtering). Morphological operators.
- Introduction to image analysis and computer vision. Segmentation, edge detection, objet detection, image comparison.

Matière examinée / subjects examined		Traitement d'images I			
Session	PRI	Coefficient / Crédits ECTS	3	Forme de l'examen / Form of examination	Ecrit

Titre	Traitement d'images II
Title	Image processing II

Enseignant(s) / Instructor(s)	Unser Michaël: MT	Langue / Language	FR
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Microtechnique (2005-2006, Master semestre 2)	Cours: 3	1 2 3 4 5	opt
SC - EPFL (IS) (2005-2006, Master semestre 2)	Cours: 3	2 4	opt
SC - EPFL (IS) (2005-2006, Master semestre 4)	Cours: 3	2 4	opt

Objectifs:

Compréhension et maîtrise des techniques avancées du traitement d'images; imagerie mathématique. Développement en JAVA et mise en oeuvre d'algorithmes de traitement d'images; application à des exemples concrets en vision industrielle et en imagerie biomédicale.

Contenu:

- **Revue des notions fondamentales.** Transformée de Fourier multi-dimensionnelle. Convolution. Transformée en z. Filtres numériques.
- **Représentation continue de données discrètes.** Splines. Interpolation. Transformations géométriques. Décompositions multi-échelles.
- **Transformations d'images.** Transformation de Karhunen-Loève (KLT) et en cosinus (DCT). Codage JPEG. Pyramides. Décomposition en ondelettes.
- **Reconstructions à partir de projections.** Scanners aux rayons X. Transformée de Radon. Rétro-projection filtrée. Méthodes itératives.
- **Déconvolution.** Filtrage inverse et de Wiener. Formulations matricielles. Méthodes itératives.
- **Méthodes statistiques de classification.** Critères de décision. Classification Bayésienne. Estimation. Apprentissage supervisé. Coalescence.
- **Analyse d'images.** Classification de pixels.

Prérequis:

Signaux et Systèmes I et II,
Traitement d'images I (ou équivalent)

Préparation pour:

Projets de semestre et travail pratique de diplôme

Forme d'enseignement:

Cours ex cathedra, exercices et travaux pratiques sur ordinateur

Bibliographie:

Notes polycopiées

Objectives:

GOALS

Study of advanced image processing; mathematical imaging. Image processing software development and prototyping in JAVA; application to real-world examples in industrial vision and biomedical imaging.

Content:

- **Review of fundamental notions.** Multi-dimensional Fourier transform. Convolution. z-transform. Digital filters.
- **Continuous representation of discrete data.** Splines. Interpolation. Geometric transformations. Multi-scale decomposition (pyramids and wavelets).
- **Image transforms.** Karhunen-Loève transform (KLT). Discrete cosine transform (DCT). JPEG coding. Image pyramids. Wavelet decomposition.
- **Reconstruction from projections.** X-ray scanners. Radon transform. Central slice theorem. Filtered backprojection. Iterative methods.
- **Deconvolution.** Inverse and Wiener filtering. Matrix formulations. Iterative techniques (ART).
- **Statistical pattern classification.** Decision making. Bayesian classification. Parameter estimation. Supervised learning. Clustering.
- **Image analysis.** Pixel classification. Contour extraction and representation. Shape. Texture. Snakes and active contours.

Matière examinée / subjects examined		Traitement d'images II			
Session	ETE	Coefficient / Crédits ECTS	3	Forme de l'examen / Form of examination	Ctrl continu

Titre	VLSI design I
Title	VLSI design I

Enseignant(s) / Instructor(s)	Leblebici Yusuf: EL	Langue / Language	EN
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Génie électrique et électronique (2005-2006, Master semestre 1)	Cours: 2	1	obl
Microtechnique (2005-2006, Master semestre 1)	Cours: 2	2	opt
SC - EPFL (IS) (2005-2006, Master semestre 1)	Cours: 2	6	opt
SC - EPFL (IS) (2005-2006, Master semestre 3)	Cours: 2	6	opt

Objectifs:

L'objectif de ce cours est de constituer une introduction aux principes fondamentaux du développement de circuits VLSI, d'examiner les blocs constitutifs élémentaires des circuits intégrés à grande échelle, ainsi que de proposer une expérience pratique de développement au moyen d'outils de design professionnels.

Contenu:

1. Introduction aux concepts de base, techniques de développement VLSI
2. Principales étapes du flot de développement VLSI - design hiérarchique
3. Technologie de fabrication CMOS, limitations, origines des règles de design, problèmes liés au développement en technologies fortement submicroniques (VDSM)
4. Développement par dessin des plans de masque
5. Parasites d'interconnexion RC, leur influence sur les performances
6. Technique de développement VLSI haute performances
Porte à plusieurs entrées, et portes complexes
Optimisation de la profondeur logique
Optimisation de la dissipation de puissance
7. Développement de sous-systèmes et architectures arithmétiques
Additionneurs à propagation de retenue
Additionneurs "Carry Lookahead"
Additionneurs "Carry Select"
Multiplieurs série/parallèle
Multiplieurs à matrice parallèle
Registres à décalage
8. Règles de développement pour circuits dédiés
Développement de circuits asynchrones
Techniques d'amplification d'horloge
Techniques de pipelining
Développement VLSI faible consommation
Génération et distribution des signaux d'horloge

Préparation pour:

Conception VLSI II

Forme d'enseignement:

Ex cathedra

Bibliographie:

Weste & Eshraghian, Principles of CMOS VLSI Design, 2nd edition, Notes polycopiées

Objectives:

The course objective is to introduce the fundamental principles of VLSI circuit design, to examine the basic building blocks of large-scale digital integrated circuits, and to provide hands-on design experience with professional design (EDA) platforms.

Content:

1. Introduction to basic concepts: VLSI design styles
2. Main steps of VLSI design flow - hierarchical design
3. CMOS fabrication technology, limitations, origins of design rules, very deep sub-micron (VDSM) issues
4. Full-custom layout design examples
5. RC interconnect parasitics, their influence on performance
6. High-performance CMOS design techniques
Multi-input gates and complex gates
Optimization of logic depth
Optimization of power dissipation
7. Sub-system design and arithmetic architectures
Ripple-carry adders
Carry-lookahead adders (CLAs)
Carry-select adders (CSAs)
Serial-parallel multiplier
Parallel array multipliers
Shift registers
8. ASIC design guidelines
Synchronous circuit design
Clock buffering techniques
Pipelining techniques
Low-power VLSI design
Generation and distribution of clock signals

Prerequisite for:

Conception VLSI II

Form of teaching:

Ex cathedra

Matière examinée / subjects examined		VLSI design I			
Session	PRI	Coefficient / Crédits ECTS	2	Forme de l'examen / Form of examination	Ecrit

Titre	VLSI design II
Title	VLSI design II

Enseignant(s) / Instructor(s)	Leblebici Yusuf: EL	Langue / Language	EN
Programme(s) Période(s)	Nombre d'heures / Hours per week	Spéc. / filière / orient	Type cours
Génie électrique et électronique (2005-2006, Master semestre 2)	Cours: 2	1	obl
Microtechnique (2005-2006, Master semestre 2)	Cours: 2	2	opt
SC - EPFL (IS) (2005-2006, Master semestre 2)	Cours: 2	6	opt
SC - EPFL (IS) (2005-2006, Master semestre 4)	Cours: 2	6	opt

Objectifs:

Le but de ce cours est de familiariser les étudiants au développement VLSI de circuits par l'usage d'outils permettant l'automatisation de phases de conception de circuits électroniques. Plusieurs blocs fonctionnels seront développés dans le cadre d'exercices pratiques ; de même, des exemples d'intégration au niveau système seront démontrés.

Contenu:

1. Introduction à la CAO pour la VLSI
Revue des systèmes CAO. Flot de conception automatique. Approches descendante et montante. Aspects pratiques de l'utilisation d'outils CAO.
2. Conception physique automatique
Partitionnement au niveau système et plan de masses. Partitionnement logique. Algorithmes de placement de modules. Algorithmes de routage global et de détail. Methodologies de compaction. Conception de layout dirigée par les performances.
3. Projets de conception
Les étudiants participeront à une série d'exercices collectifs de conception, à l'occasion desquels chaque groupe se verra assigné une tâche à terminer en 3 à 4 semaines. La difficulté des tâches assignées augmentera de façon progressive, conduisant à la réalisation de système monopuce (system-on-chip) au terme du semestre.

Prérequis:

Conception VLSI - I, Modélisation des systèmes numériques intégrés.

Forme d'enseignement:

Ex cathedra / exercices pratiques

Bibliographie:

Notes polycopiées

Objectives:

This course aims to familiarize the students with the design of very large-scale integrated (VLSI) circuits, using dedicated electronic design automation tools. Several functional blocks will be designed in practical exercises, and examples of system level integration will be shown.

Content:

1. Introduction to VLSI CAD
Overview of CAD systems. Concept of automated design flow. Top-down and bottom-up design approaches. Practical aspects of using CAD systems in design.
2. Physical Design Automation
System-level partitioning and floor-planning. Logic partitioning. Module placement algorithms. Global and detailed routing algorithms. Design compaction methodologies. Performance-driven physical layout design.
3. Design Projects
The students will participate in a series of collaborative design exercises where each project group is assigned a task, to be completed in 3-4 weeks. The complexity of the design assignments will increase progressively, leading up to system-on-chip (SoC) realization by the end of the semester.

Required prior knowledge:

Conception VLSI - I, Modélisation des systèmes numériques intégrés.

Form of teaching:

Ex cathedra / exercices pratiques

Matière examinée / subjects examined		VLSI design II			
Session	ETE	Coefficient / Crédits ECTS	2	Forme de l'examen / Form of examination	Ecrit

INDEX PAR COURS

Cours	Enseignant	Semestre	Page
Advanced compiler construction.....	Schinz M.	M2.....	87
Advanced computer architecture.....	lenne P.	M2.....	88
Advanced computer graphics.....	Thalmann D.	M2.....	89
Advanced databases.....	Spaccapietra S.	M1, M3.....	90
Advanced digital design	Sanchez E.	M2.....	91
Algèbre linéaire	Cibils M.	B1.....	41
Algorithmique	Shokrollahi A.	B4.....	57
Analyse I, II.....	Bachmann O.	B1, B2.....	42, 44
Analyse III.....	Liebendörfer Ch.	B3.....	58
Analyse de données génétiques	Morgenthaler S.	M1, M3.....	133
Analyse numérique.....	Burman E.	B4.....	59
Analysis I, II.....	Semmler K.	B1, B2.....	43, 45
Architecture des ordinateurs I, II.....	lenne P.	B3, B4.....	60, 61
Automatique I	Longchamp R.	M1, M3.....	134
Automatique II	Longchamp R. / Gillet D.	M2.....	135
Bases de données.....	Spaccapietra S.	B4.....	62
Color imaging	Suesstrunk S.	M1, M3.....	136
Color reproduction.....	Hersch R.D.	M2.....	92
Compiler construction.....	Odersky M.	B5.....	63
Complex circuits.....	Piguet Ch. / Beuchat R.	M1, M3.....	93
Computational genomics.....	Galisson F.	M1, M3.....	94
Computational processing of textual data	Rajman M. / Chappelier J.-C.	M2.....	95
Computer-aided verification.....	Henzinger Th.	M1, M3.....	96
Computer graphics	Thalmann D.	B5.....	64
Computer-supported cooperative work	Dillenbourg P.	M1, M3.....	97
Computer vision	Fua P.	M2.....	98
Concurrence.....	Sandoz A.	B5.....	65
Cryptography and security	Oechslin P. / Vaudenay S.	M1, M3.....	99
Design technologies for intergrated systems.....	De Micheli G.	M1, M3.....	100
Distributed algorithms.....	Schipper A.	M1, M3.....	101
Distributed information systems	Aberer K.	M1, M3.....	102
Dynamical system theory for engineers	De Feo O. / Hasler M.	M1, M3.....	103
E-Business	Pigneur J.-Y.	M1, M3.....	137
Electronique I, II	Zysmann E.	B1, B2.....	46, 47
Embedded systems.....	Beuchat R.	M2.....	104
Foundation of image science.....	Fua P.	M1, M3.....	105
Genetic and develop. computing architecture	Temepesti G. / Ijspeert A.	M1, M3.....	106
Gestion de production I, II	Glardon R.	M1, M2.....	138, 139
Graphes et réseaux I, II	De Werra D.	M1, M2.....	107, 108
Hardware systems modeling I, II	Vachoux A.	M1/M3, M2.....	140, 141
Human computer Interaction	Pu P.	M2.....	109
Identification et commande I.....	Karimi A.	M1, M3.....	142
Identification et commande II.....	Longchamp R. / Karimi A.	M2.....	143
Industrial automation	Kirrmann H.	M2.....	110
Infchimie.....	Roethlisberger U./ Tavernelli I.	M2.....	144
Informatique du temps réel.....	Decotignie J.-D.	B5.....	66
Informatique théorique I, II	Bartholdi L.	B1, B2.....	48, 49
Intelligence artificielle	Faltings B.	B6.....	67
Intelligent agents	Faltings B.	M1, M3.....	111
Introduction au marketing/finance	Wegmann A. / Schwab J.-M.	B6.....	68
Introduction aux syst. informatiques.....	Sanchez E.	B1.....	50
Mathematical modelling of DNA I	Maddocks J.	M2.....	145
Mathematical modelling of DNA II	Maddocks J.	M2.....	146
Mathématiques discrètes	Hêche J.-F.	B5.....	69
Mécatronique.....	Colombi S.	M2.....	147
Mobile Networks.....	Hubaux J.-P.	M1, M3.....	112
Modelling the immune system	Kraehenbuehl J.-P./Le Boudec J.-Y./Martinoli A.	M1, M3.....	113
Modèles stoch. pour les commun.....	Thiran P. / Dousse O.	M1, M3.....	148
Models of biolo. sensory-motor systems	Ijspeert A.	M1, M3.....	114
Multimedia documents	Vanoirbeek Ch.	M2.....	115
Optimisation I - (spécialisation no 4)	Bierlaire M.	M1, M3.....	149
Optimisation II - (spécialisation no 4)	Prodon A.	M2.....	150
Parral. de prog. sur grappes de PC.....	Hersch R. D.	M2.....	116
Pattern classification an Machine Learning	Gerstner W. / Hasler M.	M2.....	117

INDEX PAR COURS

Cours	Enseignant	Semestre	Page
Performance evaluation	Le Boudec J.-Y.	M2	118
Périphériques	Hersch R. D. / Gerlach S.	M2	119
Physique générale I.....	Meylan G.	B2.....	51
Physique générale II.....	Félix Ch.	B3.....	70
Probabilité et statistique I, II	Mountford Th.	B3, B4	71, 72
Programmation I.....	Sam J.	B1.....	52
Programmation II.....	Vacat.....	B2.....	53
Programmation III.....	Chappelier J.-C.	B3.....	73
Programmation IV	Odersky M.	B4.....	74
Programmation internet.....	Petitpierre C.	B4.....	75
Projet d'informatique	Divers enseignants	M1 ou M2 ou M3..	120, 121
Projet génie logiciel I, II	Hulaas J. / Petitpierre C.	B5, B6	76, 77
Projet STS (hiver)	Coray G.	M1	122
Projet STS (été)	Galland B.	M2	123
Real-time embedded systems	Beuchat R.	M1, M3	124
Real-time programming.....	Decotignie J.-D.	M1, M3	125
Real-time systems.....	Decotignie J.-D.	M2	126
Recherche opérationnelle	Hêche J.-F.	B6	78
Recherche opérationnelle - (spéc. no 4)	Spada M.	M1, M3	151
Réseaux de neu. et mod. biologique	Gerstner W.	M2	152
Réseaux informatiques.....	Grossglauser M.	B6	79
Software engineering	Baar Th.	B5	80
Swarm intelligence	Martinoli A.	M1, M3	127
Systèmes d'exploitation.....	Sandoz A.	B6.....	81
Systèmes logiques	Sanchez E.	B2.....	54
Systèmes multivariables I, II	Muellhaupt Ph.	M1/M3, M2	153, 154
Systèmes répartis.....	Schiper A.	B6	82
Theoretical computer science III.....	Henzinger Th.	B3	83
Théorie de l'information.....	Chappelier J.-C.	B6.....	84
Traitement automatique de la parole.....	Boulevard H.....	M1, M3	128
Traitement d'images I, II.....	Unser M.	M1/M3, M2	155, 156
Virtual Relatiry	Thalmann D.	M2	129
VLSI design I, II.....	Leblebici Y.	M1/M3, M2	157, 158

INDEX PAR ENSEIGNANTS

Enseignant	Cours	Semestre	Page
Aberer K.	Distributed information systems.....	M1, M3	102
Baar Th.	Software engineering.....	B5	80
Bachmann O.	Analyse I, II	B1, B2	42, 44
Bartholdi L.	Informatique théorique I, II.....	B1, B2	48, 49
Beuchat R.	Complex circuits	M1, M3	93
Beuchat R.	Embedded systems	M2	104
Beuchat R.	Real-time embedded systems	M1, M3	124
Bierlaire M.	Optimisation I - (spécialisation no 4).....	M1, M3	149
Boullard H.	Traitement automatique de la parole.....	M1, M3	128
Burman E.	Analyse numérique.	B4	59
Chappelier J.-C.	Computational processing of textual data.....	M2	95
Chappelier J.-C.	Programmation III	B3	73
Chappelier J.-C.	Théorie de l'information.....	B6	84
Cibils M.	Algèbre linéaire	B1	41
Coray G.	Projet STS (hiver)	M1	122
Colombi S.	Mécatronique	M2	147
De Feo O.	Dynamical system theory for engineers.....	M1, M3	103
De Micheli G.	Design technologies for intergrated systems	M1, M3	100
De Werra D.	Graphes et réseaux I, II	M1, M2	107, 108
Decotignie J.-D.	Informatique du temps réel.....	B5	66
Decotignie J.-D.	Real-time programming.....	M1, M3	125
Decotignie J.-D.	Real-time systems	M2	126
Dillenbourg P.	Computer-supported cooperative work.....	M1, M3	97
Divers enseignants	Projet d'informatique	M1 ou M2 ou M3	120, 121
Dousse O.	Modèles stoch. pour les commun.	M1, M3	148
Faltings B.	Intelligence artificielle	B6	67
Faltings B.	Intelligent agents	M1, M3	111
Félix Ch.	Physique générale II.....	B3	70
Fua P.	Computer vision	M2	98
Fua P.	Foundation of image science.....	M1, M3	105
Galisson F.	Computational genomics.....	M1, M3	94
Galland B.	Projet STS (été)	M2	123
Gerlach S.	Périphériques	M2	119
Gerstner W.	Pattern classification an Machine Learning	M2	117
Gerstner W.	Réseaux de neu. et mod. biologique	M2	152
Gillet D.	Automatique II	M2	135
Glardon R.	Gestion de production I, II	M1, M2	138, 139
Grossglauser M.	Réseaux informatiques.....	B6	79
Hasler M.	Dynamical system theory for engineers	M1, M3	103
Hasler M.	Pattern classification an Machine Learning	M2	117
Hêche J.-F.	Mathématiques discrètes.....	B5	69
Hêche J.-F.	Recherche opérationnelle.....	B6	78
Henzinger Th.	Computer-aided verification.....	M1, M3	96
Henzinger Th.	Theoretical computer science III.....	B3	83
Hersch R. D.	Périphériques	M2	119
Hersch R. D.	Parral. de prog. sur grappes de PC.....	M1, M3	116
Hersch R.D.	Color reproduction	M2	92
Hubaux J.-P.	Mobile Networks	M1, M3	112
Hulaas J.	Projet génie logiciel I, II	B5, B6	76, 77
lenne P.	Architecture des ordinateurs I, II.....	B3, B4	60, 61
lenne P.	Advanced computer architecture	M2	88
Ijspeert A.	Genetic and develop. computing architecture	M1, M3	106
Ijspeert A.	Models of biolo. sensory-motor systems	M1, M3	114
Karimi A.	Identification et commande I, II.....	M1/M3, M2	142, 143
Kirrmann H.	Industrial automation	M2	110
Kraehenbuehl J.-P.	Modelling the immune system	M1, M3	113
Le Boudec J.-Y.	Modelling the immune system	M1, M3	113
Le Boudec J.-Y.	Performance evaluation.....	M2	118
Leblebici Y.	VLSI design I, II	M1/M3, M2	157, 158
Liebendörfer Ch.	Analyse III	B3	58
Longchamp R.	Identification et commande II.....	M2	143
Longchamp R.	Automatique I, II	M1/M3, M2	134, 135
Maddocks J.	Mathematical modelling of DNA I	M2	145
Maddocks J.	Mathematical modelling of DNA II	M2	146
		M1, M3	113

INDEX PAR ENSEIGNANTS

Enseignant	Cours	Semestre	Page
Martinoli A.	Modelling the immune system	M1, M3	127
Martinoli A.	Swarm intelligence	B2	51
Meylan G.	Physique générale I	M1, M3	133
Morgenthaler S.	Analyse de données génétiques	B3, B4	71, 72
Mountford Th.	Probabilité et statistique I, II	M1/M3, M2	153, 154
Muellhaupt Ph.	Systèmes multivariés I, II	B5	63
Odersky M.	Compiler construction	B4	74
Odersky M.	Programmation IV	M1, M3	99
Oechslin P.	Cryptography and security	B5, B6	76, 77
Petitpierre C.	Projet génie logiciel I, II	B4	75
Petitpierre C.	Programmation internet	M1, M3	137
Pigneur J.-Y.	E-Business	M1, M3	93
Piguet Ch.	Complex circuits	M2	150
Prodon A.	Optimisation II - (spécialisation no 4)	M2	109
Pu P.	Human computer Interaction	M2	95
Rajman M.	Computational processing of textual data	M2	144
Roethlisberger U.	Infochimie	B1	52
Sam J.	Programmation I	M2	91
Sanchez E.	Advanced digital design	B1	50
Sanchez E.	Introduction aux syst. informatiques	B2	54
Sanchez E.	Systèmes logiques	B5	65
Sandoz A.	Concurrence	B6	81
Sandoz A.	Systèmes d'exploitation	M2	87
Schinz M.	Advanced compiler construction	M1, M3	101
Schipper A.	Distributed algorithms	B6	82
Schipper A.	Systèmes répartis	B6	68
Schwab J.-M.	Introduction au marketing/finance	B1, B2	43, 45
Semmler K.	Analysis I, II	B4	57
Shokrollahi A.	Algorithmique	M1, M3	90
Spaccapietra S.	Advanced databases	B4	62
Spaccapietra S.	Bases de données	M1, M3	151
Spada M.	Recherche opérationnelle - (spécialisation no 4)	M1, M3	136
Suesstrunk S.	Color imaging	M2	144
Tavernelli I.	Infochimie	M1, M3	106
Temepesti G.	Genetic and develop. computing architecture	M2	89
Thalmann D.	Advanced computer graphics	B5	64
Thalmann D.	Computer graphics	M2	129
Thalmann D.	Virtual Reality	M1, M3	148
Thiran P.	Modèles stoch. pour les commun.	M1/M3, M2	155, 156
Unser M.	Traitement d'images I, II	B2	53
Vacat	Programmation II	M1/M3, M2	140, 141
Vachoux A.	Hardware systems modeling I, II	M2	115
Vanoirbeek Ch.	Multimedia documents	M1, M3	99
Vaudenay S.	Cryptography and security	B6	68
Wegmann A.	Introduction au marketing/finance	B1, B2	46, 47
Zysmann E.	Electronique I, II		