CARAT Homepage e-Mail

INTRODUCTION TO CARAT

CARAT is a compilation of various small programs written in C, which can solve certain problems in crystallography. It is distributed via

Lehrstuhl B für Mathematik
RWTH-Aachen
Prof. Plesken
Templergraben 64
52064 Aachen
Germany
email: carat AT momo.math.rwth-aachen.de

Contents

Programs

There are three categories of programs in CARAT, regarding their importance.

Every program should give some online-help if used with the option -h.

Most frequently used programs

Here a short list of the most important executables is given.

Program/Synonyms **Short description**

Aut_grp	Calculates the automorphism group of one or more quadratic forms.
Bravais_catalog	Provides a list of all Bravais groups up to degree 6. Bravais_catalog is a synonym for Datei.
Bravais_grp	Calculates the Bravais group of a finite unimodular group
Bravais_inclusions	Outputs Bravais subgroups/supergroups for a given Bravais group.
Bravais_type	Calculates the family symbol of a finite unimodular group. Also calculates an equivalent group in the catalog of Bravais groups. Note that Bravais_type is nothing else then Symbol -i.
Datei	Provides a list of all Bravais groups up to degree 6. Datei is a synonym for Bravais_catalog.
Extensions	Calculates all non-isomorphic extensions of a finite unimodular group with a given lattice. Extensions is a synonym for Vector_systems
Form_space	Calculates the space of invariant forms of a unimodular group.
Graph	Calculates the "graph of inclusions" for a given geometric class.
Is_finite	Decides finiteness of a given subgroup of GL_n(Z). Calculates the order in case the group is finite.
KSubgroups	Calculates the maximal klassengleich subgroups of a spacegroup for some prime-power index.
KSupergroups	Calculates the maximal klassengleich supergroups of a spacegroup for some prime-power index.
Name	Give a space group a name, ie. calculate a string which describes the isomorphism type uniquely, cf. Reverse_name.
Normalizer	Calculates the Normalizer in GL_n(Z) of a given finite unimodular group.
Orbit	Fairly general implementation of the orbit/stabilizer algorithm.
Order	Calculates the order of a given finite subgroup of GL_n(Q).
Q_catalog	Provides a list of all Q_classes up to degree 6.
QtoZ	Splits a Q-class into Z-classes.
Reverse_name	Constructs a space group with given name, and check whether the name is valid, cf. Name.
	Transforms the generators of a space group to a prescribed linear

Same_generators	part.
Symbol	Calculates the family symbol of a finite unimodular group. Also calculates an equivalent group in the catalog of Bravais groups. Note that Symbol -i is nothing else then Bravais_type
Torsionfree	Decides whether a given space group is torsion free. WARNING: The program assumes the translation subgroup to be Z^n.
TSubgroups	Calculates the maximal translationengleich subgroups of a space group up to conjugation in this space group or under the affine normalizer.
TSupergroups	Calculates the minimal translationengleich supergroups of a space group up to conjugation under the affine normalizer of this space group.
Vector_systems	Calculates all non-isomorphic extensions of a finite unimodular group with a given lattice. Vector_systems is a synonym for Extensions
Z_equiv	Decides whether two given finite unimodular groups are conjugated in GL_n(Z).

Less frequently used programs

We continue with given the name of some additional functions which the user might find useful.

Program/Synonyms	**Short description**
Bravais_equiv	Decides whether the Bravais groups of two given finite unimodular groups are conjugated
Conj_bravais	Conjugates a Bravais group with a given matrix
Extract	Tool to get from space groups to point groups and vice versa.
Idem	Calculates (rational) central primitive idempotents of the enveloping algebra of a given matrix group.
Invar_space	Form_space. Is much faster than this, but uses some random methods.
Isometry	Calculates an isometry of with respect to tuples of bilinear forms.
Long_solve	Solves linear systems of equations using multiple precision integers.

Mink_red	The Minkowski reduction of bilinear forms. Gives very good results, but use Pair_red before.
Pair_red	Pair reduction of bilinear forms. Very fast.
Presentation	Calculates a presentation of a finite soluble subgroup of GL_n(Z)
Red_gen	Tries to reduce the number of elements of a generating set of a finite matrix group.
Rein	Purifies a lattice.
Rform	Mostly used for finding a positive definite G-invariant form or a finite unimodular group G.
Scpr	Calculates scalar products w.r.t a given form.
Short	Calculates short vectors of a given positive definite symmetric form.
Shortest	Shortest vectors of a given positive definite symmetric quadratic form.
Signature	Sylvester type of a quadratic form. In particular it decides whether a given form is positive definite.
Standard_affine_form	Standard_affine_form is just Extract -t
Sublattices	Find G-invariant sublattices of Z^n. Note that this is a dualisation of finding centerings. ZZprog is a synonym for ZZProg.
Tr_bravais	Transposes a finite unimodular group.
Zass_main	Calculates H^1(G,Q^n/Z^n) for a given finite unimodular group.
ZZprog	Find G-invariant sublattices of Z^n. Note that this is a dualisation of finding centerings. ZZprog is a synonym for Sublattices.

Programs seldom used and those for debugging

The remaining functions are merely of debugging and processing the results, nevertheless an experienced user might calculate relevant data with them.

t

Program/Synonyms	**Short description**
Add	Adds matrices
Con	Conjugates matrices
Conjugated	Decides whether two groups are conjugate under third group.

Conv	Converts CARAT input-file (matrix_TYP) into GAP and Maple format.
Elt	An elementary divisors algorithm.
First_perfect	Find G-perfect forms.
Form_elt	Elementary divisors of the trace bilinear form of a finite unimodular group. Useful for distinguishing Bravais groups.
Formtovec	Writes a given form as linear combination of others.
Full	Outputs given matrices in a full form, which might be easier to edit.
Gauss	An implementation of Gauss's algorithm.
Inv	Inverts matrices.
Kron	Kronecker product of matrices.
Ltm	Inverse to Mtl.
Minpol	Minimal polynomial of integral matrices.
Modp	Takes all entries of a matrix mod p a prime.
Mtl	Writes matrices in lines.
Mul	Multiplies matrices.
Normalizer_in_N	Calculates the normalizer of a finite group in a second one.
Normlin	Calculates for each matrix A in 'file2' a matrix X with the property that $\mathrm{Sum}_j X_{i,j} F_j = A^{tr} F_j A$ with F_j in 'file1'.
P_lse_solve	Solves a system of equations modularly.
Pdet	Determinant of a matrix mod p.
Perfect_neighbours	Gives the perfect neighbours of a given G-perfect form.
Polyeder	
Rest_short	
Scalarmul	Multiplies matrices with rational number.
Short_reduce	
Simplify_mat	Divides all entries of a matrix by their greatest common divisor.
Tr	Transposes matrices.

Trace	Trace of matrices.
Trbifo	Trace bilinear form of a finite unimodular group.
Vectoform	Calculates a linear combintion of forms.
Vor_vertices	

There is also an alphabetical program list!

Files for in/output

In principle CARAT does know two different file formats in which the in/output takes place. The first and most basic one is matrix_TYP and the second and most frequent one is bravais_TYP.

matrix_TYP

The format of a single matrix for CARAT is a preceding line

```
NxM   % comment
```

telling the programs to read a matrix with N lines and M columns. Spaces, tabs and so on are ignored, and so is everything behind % in the same line.
Following this line the program will read N*M integers, which represent the matrix ROW BY ROW, regardless of spaces, cr, tabs and so on. Therefore all the following examples stand for the same matrix.

```
3x4   % most natural way to put it
1   2   3   4
5   6   7   8
9  10  11  12

3x4   % even this
1  2  3  4  5  6  7  8  9  10  11  12

3x4
1  2  3  4  5  6
7  8  9  10  11  12
```

Furthermore there are some abbreviations allowed, which deal with square matrices and those having symmetries. In the header line of a matrix N is equivalent to NxN. The following examples describe the same matrix:

```
2x2
1  2
3  4
```

```
2
1 2
3 4
```

Again, formating characters are ignored. Coming to matrices which obey symmetries CARAT follows the konvention that Nx0 means an symmetric N by N matrix, of which program just will read the lower triangular. Note that all the following examples have the same meaning:

```
2
1 2
2 1

2x0
1
2 1

2x0
1 2 1
```

The last abbreviation are meant for diagonal matrices, which are Nd1 for a N by N diagonal matrix, of which program will read N diagonal entries, and Nd0 for a N by N scalar matrix, of which only the defining scalar is read. Again a couple of outputs meaning the same thing should make it clear.

```
3x3
2 0 0
0 2 0
0 0 2

3d1
2 2 2

3d0
2
```

Most programs will read more than one matrix. Therefore a matrix_TYP normaly constits of a preceding line of the form #A , where A is the number of matrices to be read. In the next example we give a matrix_TYP consisting of 2 matrices (which generate a group isomorphic to S_4, the permutation group on four letters).

```
#2
3    % presentation for a transposition
0 1 0
1 0 0
0 0 1

3    % presentation of a 4-cycle
 0  1  0
 0  0  1
-1 -1 -1
```

rational matrices

The way CARAT presents rational matrices is to divide the whole thing by an integer:

```
3/2    % divide the whole matrix by 2
1 2 3
4 5 6
7 8 9
```

A matrix discribing a presentation

This is a slight abuse of notation, but nevertheless a matrix_TYP in CARAT can discribe a finitely presented group.

A single line of this matrix will present a relation fullfilled by the generators of the group, and the biggest entry in modulus will be the number of generators. Words in the free group translate in the obvious way to a line of a matrix, therefore we just give a couple of ways of presenting the group $V_4 = C_2$ X C_2. To make the matrix rectangular, fill the shorter rows with zeroes.

```
3x4       % we will need 3 relations, the longest of which will have 4 entries
1 1  0  0
2 2  0  0
1 2 -1 -2
```

The three lines read: $x_1x_1 = 1$, $x_2x_2 = 1$, $x_1x_2x_1^{-1}x_2^{-1} = 1$. Of course there are various ways to put it, like

```
3x4
1 1 0 0
2 2 0 0
1 2 1 2
```

or

```
3x4
1 1 2 2
1 1 0 0
1 2 1 2
```

bravais_TYP

A bravais_TYP in CARAT is used to decribe a group generated by matrices together with additional information like their normalizers and a basis for the space of invariant forms.
The bravais_TYP consists of a header line, which tells the program how many matrices to be read, and how to interpret them.
This header line takes the following form:

```
#gA fB ZC nD cE % just a comment
```

where A, B, C, D and E are natural numbers. It advises the program to read A + B + C + D + E matrices, where A matrices are meant to generate the group, the next B matrices form an integral basis of the space of fixed forms, followed by C matrices giving so called "centerings". The program proceeds in reading D matrices which generate the normalizer of the group (modulo the group generated by the group and its centralizer), and E matrices which generate the centralizer of the group.

Note: It is possible to ommit any of the records which discribe generators, the space of forms and so on, but it is NOT possible to switch components.

The next example gives a bravais_TYP:

```
#g2 f1 n3 % group with complete normalizer
3       % generator
 0 1 0
 1 0 0
 0 0 1
3       % generator
  0  1  0
  0  0  1
 -1 -1 -1
3x0     % invariant form
 2
 1 2
 1 1 2
3       % generator of normalizer
  1  1 1
  0 -1 0
 -1  0 0
3       % generator of normalizer
 -1 0  0
  0 0 -1
  1 1  1
3x0     % generator of normalizer
 1
 0 0
 0 1 0
2^3  * 3^1   = 24 % order of the group
```

Note that the order of the group is given at the end, and that it is factorized. Some programs are using this line. These programs assume the order given to be right.

Examples

1. List all names of Bravais groups of degree 4.
2. How do the Z-classes in the Q-class of a given group distribute into their Bravais flocks?
3. Determine the maximal finite subgroups of $GL_5(Z)$
4. Find all space groups with a given point group and decide for which superlattices each extension splits.
5. Do two given matrices generate a space group?

6. Find the dual pairs of Bravais groups in family 1;1;1;1
7. Is the normalizer of a given group finite?
8. Find all Bravaisgroups of degree 6 consisting of permutation matrices and their negatives only.
9. Do two given groups have Z-equivalent copies which lie in a finite unimodular group?
10. Find the stabilizer of a sublattice in the Bravais group of the unit form $F=I_6$.
11. Find the fundamental group of the Handtsche-Wendt manifold!
12. Find all Z-classes, affine classes and torsion free space groups in a given Q-class.
13. Calculate the "graph of inclusions" for a given geometric class.
14. Calculate minimal/maximal klassengleich/translationengleich super-/sub-groups of a given space group up to conjugation under the affine normalizer of this group.

Remarks

If you find any Bug in CARAT, we would be pleased to hear from you. Please send us a copy of the file you produced the error with, and a log from the things you did with it.
A short explanation why you encounter the result (if you got any) to be wrong would be helpful as well.

We would also be pleased to hear from you, if you find any errors or misprints in this introduction.

CARAT Homepage e-Mail Up

last change: 07.01.2003