

Exemples d'exercices - Semaine 11

1 Mémoire cache

Le produit matriciel de deux matrices 2x2

$$A = \begin{Bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{Bmatrix} \text{ et } B = \begin{Bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{Bmatrix}$$

est défini comme une matrice 2x2

$$P = \begin{Bmatrix} p_{11} & p_{12} \\ p_{21} & p_{22} \end{Bmatrix} \text{ où } \begin{aligned} p_{11} &= (a_{11}b_{11} + a_{12}b_{21}) & p_{12} &= (a_{11}b_{12} + a_{12}b_{22}) \\ p_{21} &= (a_{21}b_{11} + a_{22}b_{21}) & p_{22} &= (a_{21}b_{12} + a_{22}b_{22}) \end{aligned} \text{ et}$$

Supposons que le processeur dispose d'assez de registres pour contenir tous les résultats intermédiaires. Supposons aussi que chaque élément matriciel occupe un bloc de 4 octets, que la mémoire cache puisse contenir un maximum de six blocs de 4 octets, et qu'elle est gérée par l'algorithme en tuyau FIFO (First-In-First-Out). Avec FIFO, quand la cache est pleine et qu'il faut y remplacer un bloc, on enlève non pas le bloc le moins récemment utilisé (comme le ferait le classique algorithme LRU vu au cours) mais le bloc le moins récemment importé dans la mémoire cache (qu'il ait été utilisé récemment ou pas). La mémoire cache se comporte donc un tuyau qu'on remplit par un bout et qui se vide par l'autre.

Lorsqu'un programme calcule un élément de P, il charge les éléments de A et B de gauche à droite selon les équations ci-dessus. Par exemple, pour p_{11} il charge les éléments dans l'ordre $a_{11}, b_{11}, a_{12}, b_{21}$.

On peut calculer les éléments de P dans l'ordre

(a) $p_{11}, p_{12}, p_{21}, p_{22}$

ou dans l'ordre

(b) $p_{11}, p_{21}, p_{12}, p_{22}$

Le nombre de défauts de cache causé par (a) est-il

< = ou >

au nombre de défauts de cache causé par (b) ?