

# Sujets de TD n° 5

---

## Exercice 1.

---

On considère un problème d'une certaine taille. Un programme séquentiel passe 6% de son temps d'exécution dans les opérations d'entrée/sortie. Ce programme est parallélisable mais les opérations d'entrées-sorties ne peuvent être exécutées que par un seul cœur.

Quel est le nombre minimal de cœurs pour que le programme parallèle produise une accélération (*speedup*) de 10 par rapport au programme séquentiel ?

## Exercice 2.

---

Un programme a une accélération de 9 sur 10 processeurs.

Quelle est la fraction maximale du temps d'exécution séquentiel qui peut correspondre à des opérations purement séquentielles ?

## Exercice 3.

---

*NB : cet exercice est inspiré de « Isoefficiency : Measuring the Scalability of Parallel Algorithms and Architectures », A. Grama, A. Gupta and V. Kumar, IEEE Parallel & Distributed Technology, août 1993.*

Soit un programme traitant un problème de taille  $W$  ( $W$  est le nombre total d'opérations à réaliser). On suppose que le temps d'exécution d'une opération est  $t_c$ .

### Question 1.

Soit  $T_o(p)$  le coût total de la parallélisation de ce programme pour  $p$  processeurs (communications, synchronisations, etc. — on considère que ce coût est équitablement réparti entre les processeurs). On suppose que le code peut être entièrement parallélisé. Quel est le speedup  $S(p)$  de l'exécution parallèle du programme sur  $p$  processeurs ?

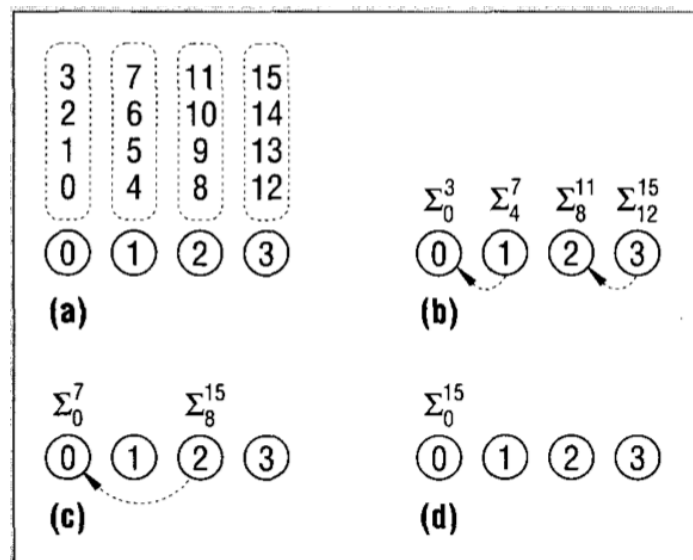
### Question 2.

L'efficacité de la parallélisation  $E$  est définie comme le speedup divisé par le nombre de processeurs. Donner l'expression de  $E(p)$ .

Application : on considère un programme qui calcule la somme des éléments d'un vecteur. Si le vecteur comporte  $n$  éléments, le programme comporte  $n$  opérations (lecture d'un élément, ajout à la somme, instructions de gestion de la boucle de parcours du vecteur). On suppose que le temps d'exécution d'une opération est de une unité de temps.

### Question 3.

Ce programme est parallélisé selon le schéma suivant (exemple pour  $n=16$  et  $p=4$ ) :



Chaque cœur  $i$  fait les calculs suivants :

```

res[i] = somme(élts locaux :  $V[i*N/P]$  à  $V[(i+1)*N/P-1]$ ) ;
j = 1 ;
tantque j <= p faire
    res[i] = res[i] + res[(i+j%P)];
    j = j*2
fintantque
  
```

On suppose qu'une communication dure une unité de temps.

Calculer le speedup et l'efficacité obtenus pour  $P$  cœurs.