



Niveau : Master I -VA

Module : Traitement parallèle de l'information

Date : 11- 05- 2024

Durée : 90 minutes

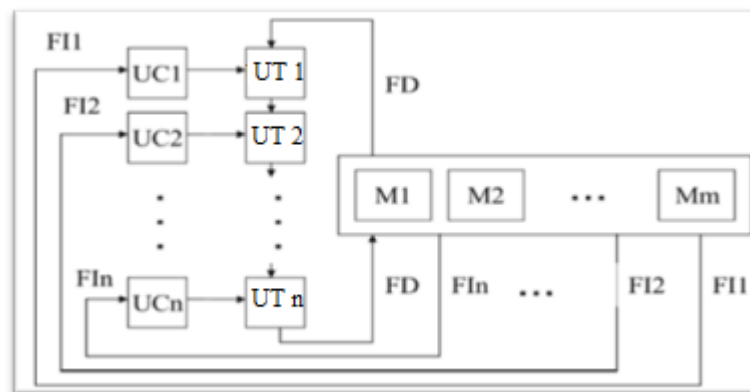
Contrôle N° 01

Exercice N 01 : (04 pts)

- 1) Donner une brève définition des mots clés suivants :
 - Barrière de synchronisation, SADM, programmation parallèle, MIPS.
- 2) Quelles sont les raisons qui nous obligent d'introduire le parallélisme?

Exercice N 02 : (04pts)

Soit le schéma suivant :



1. Donner la signification de chaque concept.
2. Donner une explication de ce schéma.
3. Ce schéma explique une des machines parallèles, citer et définir ce type de machines.
4. Donner des exemples? justifier votre réponse.

Exercice N 03 : (08 pts)

Soit deux algorithmes « A » et « B » utilisés pour résoudre, de façon séquentielle 25% et 18% respectivement, un même problème de taille n en temps de 230 et 248 secondes. On supposant que le nombre maximum de processeurs utilisés respectivement est 12 et 18 :

- 1) Calculer le **coût**, l'**accélération** et l'**efficacité**.
- 2) Quel est le type d'accélération obtenu ? justifier votre réponse.

Exercice N 04 : (04 pts)

On considère le sous-programme suivant :

```
I1: A1=A2+A3/A1;  
    For (i=1; i<=10; i++)  
    {  
I2: M[i] = A1+N[i];  
I3: N[i+1] = M[i-1]+A1;  
I4: A3 = A2*A2;
```

- 1) Donner le graphe de dépendance associé à ce sous-programme.
- 2) Citer et montrer les différentes dépendances qui existent.

o

o

o

Bon Courage

Corrigé type du contrôle N01 TP1

Exercice N 01 : (04pts)

- 1) Donner une brève définition des mots clés suivants : (03 pts)
- ☞ **Barrière de synchronisation** : permet à plusieurs threads d'attendre que tous les threads aient tous atteint un point d'exécution particulier avant qu'un thread ne continue.
 - ☞ **SADM (Single Address space, Distributed Memory)** : Mémoire distribuée, avec espace d'adressage global. Autorisant éventuellement l'accès aux données situées sur d'autres processeurs. C'est l'architecture la plus utilisée par les superordinateurs.
 - ☞ **Programmation parallèle** : est une programmation dans un langage permettant d'indiquer que certaines instructions peuvent être exécutées de façon simultanée sur différents processeurs.
 - ☞ **MIPS : (Machine Instructions Per Second)** : représente le nombre d'instructions effectuées par seconde. C'est l'unité de calcul la puissance.
- 2) Quelles sont les raisons qui nous obligent d'introduire le parallélisme? : (01 pts)
- ✓ Les besoins des applications,
 - ✓ Les limites des architectures séquentielles,
 - ✓ L'existence dans les applications de la propriété de parallélisme.

Exercice N 02 : (04pts)

1. Signification :

UC = Unité de Contrôle (0.25 pts)

UT = Unité de Traitement..... (0.25 pts)

FI = Flot d'Instructions (0.25 pts)

M = Unité Mémoire (0.25 pts)

FD = Flot de Données..... (0.25 pts)

2. Explication :

Premièrement, au niveau de chaque **processeur**, l'unité de contrôle (UC) est chargée pour faire le séquençage des instructions. Dans la deuxième étape, les différentes instructions (FI) vont aux différentes unités de traitement (UT) pour effectuer les différentes opérations. Après l'exécution des opérations, on obtient une seule donnée (FD) sauvegardé dans la **mémoire** (M) qui contient les instructions et les données qui peuvent être exécuté dans les opérations suivantes. (0.75 pts)

3.

☞ **Le type de machines** c'est MISD. (0.50 pts)

☞ **Définition:** (01 pts)

MISD “**Multiple Data, Single Instruction**”

Sont des machines vectorielles multi-processeurs :

- ✓ Peut exécuter plusieurs instructions en même temps sur la même donnée,
- ✓ Faible nombre de processeurs puissants (1 à 16)
- ✓ Mémoire partagée.
- ✓ Limite atteinte, coût important.

4. **Donner des exemples de ces machines ? justifier votre réponse.** (0.50 pts)

Théoriquement, certains auteurs considèrent que cette classe n'est pas commercialisée et d'autres auteurs considèrent le pipeline comme un schéma MISD. Par conséquent, l'architecture MISD reste un modèle théorique car il n'existe aucune machine construite sur ce modèle.

Exercice N 03 : (08pts)

Les données :

Pg (A) :	P = 12, T(n) = 230 s, $f = R = 0.25$..... (0.50 pts)
Pg (B) :	P = 18, T(n) = 248 s, $f = R = 0.18$..... (0.50 pts)

Selon l'énoncé de l'exercice, on peut appliquer les deux lois.

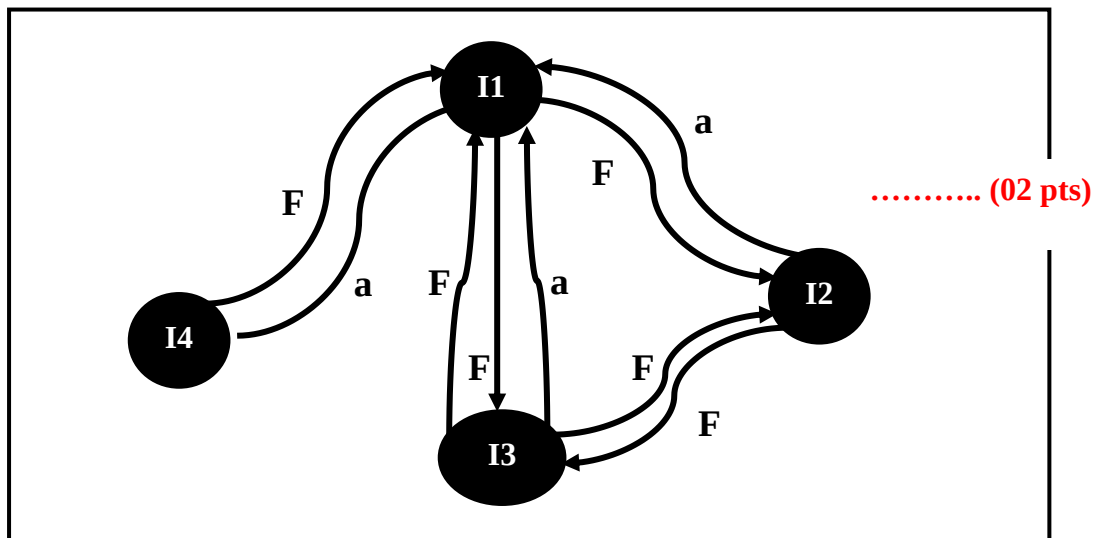
	Coût :	L'accélération :		L'efficacité :	
Loi	$C(n) = p \times T(n)$ (0.50 pts)	Amdhal : $S(p) \leq \frac{1}{f + (1-f)/p}$ (0.50 pts)	Gustafson-Barsis : $S(p) = p + (1-p) R$ (0.50 pts)	$E(n,p) = S(n,p) / p$ (0.50 pts)	
Pg A : P = 12	$C(n) = 12 \times 230$ $C(n) = 2760$ (0.50 pts)	$S(p) = 3.2$ (0.50 pts)	$S(p) = 9.25$ (0.50 pts)	Amdhal $E(n,p) = 0.26$ (0.25 pts)	Gustafson-Barsis : $E(n,p) = 0.77$ (0.25 pts)
Pg B : P = 18	$C(n) = 18 \times 248$ $C(n) = 4464$ (0.50 pts)	$S(p) = 4.43$ (0.50 pts)	$S(p) = 14.94$ (0.50 pts)	Amdhal $E(n,p) = 0.24$ (0.25 pts)	Gustafson-Barsis : $E(n,p) = 0.83$ (0.25 pts)

3) Le type d'accélération obtenu avec la justification :

Au niveau de chaque programme, et avec les deux lois on a $S(p) < P$. Donc, l'accélération obtenu est une accélération sub-linéaire. (01 pts)

Exercice N 04 : (04 pts)

1- Le graphe de dépendance associé à sous-programme.



2- Les différentes dépendances qui existent : (02 pts)

$S(I1) = E(I2) \Rightarrow DF$: Dépendance de Flot ;
 $S(I3) = E(I1) \Rightarrow DF$: Dépendance de Flot.
 $S(I2) = E(I3) \Rightarrow DF$: Dépendance de Flot.
 $E(I2) = S(I1) \Rightarrow a$: anti-dépendance.
 $E(I1) = S(I4) \Rightarrow a$: anti-dépendance.
 $S(I1) = E(I3) \Rightarrow DF$: Dépendance de Flot.
 $S(I4) = E(I1) \Rightarrow DF$: Dépendance de Flot.
 $S(I3) = E(I2) \Rightarrow DF$: Dépendance de Flot.
 $E(I3) = S(I1) \Rightarrow a$: anti-dépendance.