

Université Gaston Berger de Saint-Louis	Master Développement et de Systèmes d'Information	Année Universitaire 2017-2018
UFR DES SCIENCES APPLIQUEES ET DE TECHNOLOGIE		-----
Section Informatique		B. DIOP

## Examen de systèmes distribués

Document non autorisé !

### Exercice 1 (5 points)

Traitement de tableau à deux dimensions

Cet exemple illustre des calculs sur des éléments d'un tableau à deux dimensions. Une fonction est évaluée sur chaque élément du tableau. Le calcul sur chaque élément du tableau est indépendant des autres éléments du tableau. Le problème est intensif en calcul. Le programme sériel calcule un élément à la fois dans un ordre séquentiel. Le code de série pourrait être de la forme :

```
do j = 1, n
  do i = 1, m
    a(i, j) = fcn(i, j)
  fin do
fin do
```

Questions :

1. Est-ce que le problème peut être transformé en un problème parallèle ?
2. Proposer une subdivision du problème au cas où il est parallélisable.
3. Expliquer l'apport de la communication lorsque le problème est subdivisé en sous-problèmes.
4. Existe-t-il une dépendance des données entre les différents sous problèmes ?
5. Y a-t-il un besoin de synchronisation ? Si oui proposer un schéma simple de synchronisation, en considérant qu'un nœud central coordonne les activités du réseau.

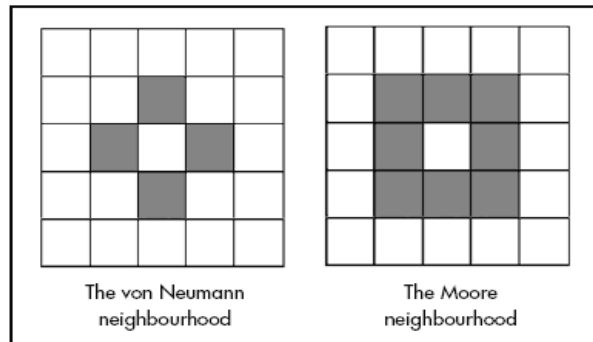
### Exercice 2 (5 points)

Automate cellulaire et jeu de la vie

Un automate cellulaire consiste en une grille régulière de « cellules » contenant chacune un « état » choisi parmi un ensemble fini d'états et qui peut évoluer au cours du temps. L'état d'une cellule au temps  $t+1$  est fonction de l'état au temps  $t$  d'un nombre fini de cellules appelé son « voisinage ».

Il y a deux types de voisinage : Von Neumann et Moore comme montrés sur la figure ci-après. Le premier a 4 voisins correspondant aux 2 cellules adjacentes verticales (en haut, en bas) et

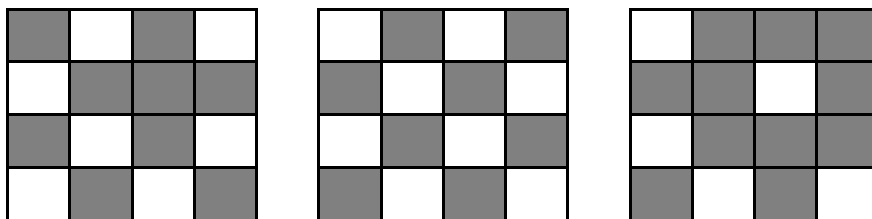
horizontales (à gauche, et à droite), alors que le voisinage de Moore compte également les cellules sur les diagonales, ce qui en fait 8 voisins.



À chaque nouvelle unité de temps, les mêmes règles sont appliquées simultanément à toutes les cellules de la grille, produisant une nouvelle « génération » de cellules dépendant entièrement de la génération précédente.

Le « [jeu de la vie](#) » est un automate cellulaire bidimensionnel où chaque cellule peut prendre deux valeurs (« 0 » en blanc ou « 1 », en noir sur la figure, mais on parle plutôt de « Vivant » ou « Mort ». L'état futur d'une cellule est déterminé par son état actuel et par le nombre de cellules vivantes parmi les cellules voisines qui l'entourent. Les règles sont les suivantes :

- Si plus de la moitié des cellules du voisinage sont dans l'état « Mort » au temps  $t$ , alors je serai dans l'état « Mort » au temps  $t+1$
- Sinon si plus de la moitié des cellules du voisinage sont dans l'état « Vivant » au temps  $t$ , alors je serai dans l'état « Vivant » au temps  $t+1$
- Sinon « Mort » ou « Vivant » au temps  $t+1$



État initial  $t=0$       État  $t=1$  (Von neumann)      État  $t=1$  (Moore)

Quelques étapes d'évolution du [jeu de la vie](#) à partir d'une configuration initiale choisie au hasard.

1. Ce problème peut-il être parallélisé ?
2. Comment le problème pourrait-il être partitionné ? Proposer un schéma de partition si oui.
3. Est-ce que la communication est nécessaire ?
4. Y a-t-il des dépendances de données ?
5. Y a-t-il besoin de synchronisation ?
6. L'équilibrage de charge sera-t-il une préoccupation ?

## Exercice 3 (10 points)

### Cours et restitution

1. A quoi consiste le parallélisme dans la programmation parallèle ?
  - a. Exécuter plusieurs programmes différents sur un même processeur en même temps
  - b. Exécuter un programme sur plusieurs processeurs différents en même temps
  - c. Exécuter de façon séquentielle les instructions d'un programme sur plusieurs processeurs différents
  - d. Exécuter de façon parallèle les instructions d'un programme sur plusieurs processeurs différents
2. Selon l'architecture des ordinateurs, nous avons deux types d'algorithmes : algorithme séquentiel, algorithme parallèle. Identifier les propriétés de chacun de ces deux types d'algorithmes en remplissant les cages vides.
  - [ ] : Un algorithme dans lequel les étapes consécutives d'instructions sont exécutées dans un ordre chronologique pour résoudre un problème.
  - [ ] : Le problème est divisé en sous-problèmes exécutés simultanément pour obtenir des résultats individuels. Ensuite, ces résultats individuels sont combinés pour obtenir le résultat final souhaitée.
3. Selon la taxinomie de Flynn, il y a quatre types d'ordinateurs. Identifier les sur la grille en reliant avec des flèches chaque type à sa description.

<u>Types</u>	<u>Description</u>
I MISD	flux d'instruction unique, et de données unique
I SIMD	flux d'instruction unique, et de données multiples
I SISD	flux d'instructions multiples, et de données unique
I MIMD	flux d'instructions multiples, et de données multiples

4. Le modèle d'un algorithme parallèle est développé en considérant une stratégie pour diviser les données et la méthode de traitement et en appliquant une stratégie appropriée pour réduire les interactions.  
Identifier parmi ces schémas ci-dessous (de (A) à (E)) le modèle concerné en encerclant la bonne réponse pour chacun des modèles ci-dessous :

Modèle parallèle de données :

(A) (B) (C) (D) (E)

Modèle de graphe des tâches

(A) (B) (C) (D) (E)

Modèle de pool de travail

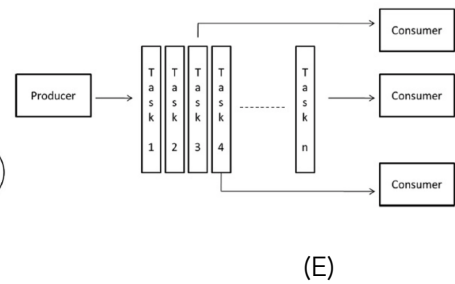
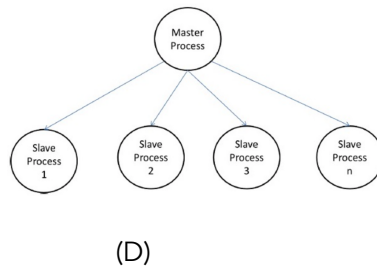
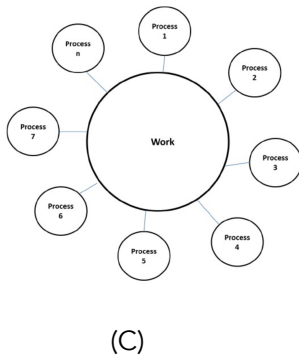
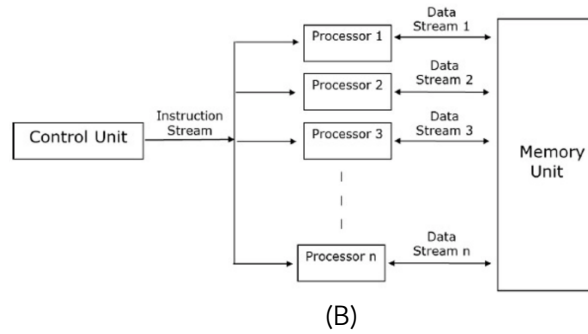
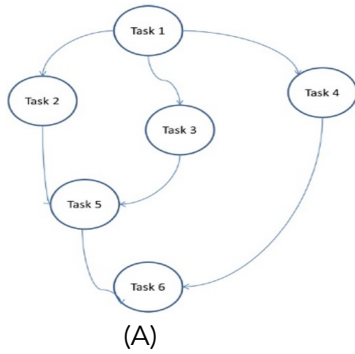
(A) (B) (C) (D) (E)

Maître esclave

(A) (B) (C) (D) (E)

Modèle de pipeline

(A) (B) (C) (D) (E)



Bonne chance !