

Important: Il sera tenu compte de la clarté des explications données par l'étudiant(e). Les barèmes sont indicatifs. Veuillez noter clairement le titre des exercices traités dans votre copie. Notez que les questions sont indépendantes. Les questions marquées d'une étoile sont les plus importantes de chaque exercice.

Documents autorisés: notes de cours et de TD.

A. Algorithmique parallèle

Exercice 1. (5 points)

On considère le problème classique suivant:

Entrées: un tableau A de n entiers.

Sortie: un tableau B de taille n tel que (les indices débutent avec 1)

$\forall i \in \{1, 2, \dots, n\}, \quad B[i] = \sum_{j=1}^i A[j]$. **Exemple:** Pour $A[] = 1, 2, 3, 4$, on obtient $B[] = 1, 3, 6, 10$.

1.1 Ecrire un algorithme EREW PRAM résolvant ce problème en utilisant n processeurs.

1.2 Quelle est le temps d'exécution de votre algorithme?

1.3 Optimiser l'algorithme en utilisant $O(n/\log n)$ processeurs. Comparer avec l'algorithme séquentiel.

Exercice 2. (5 points)

2.1 On dispose d'un tableau T de n entiers dans $\{1, 2, \dots, n\}$. Montrer comment trouver l'entier qui a le plus d'occurrences dans T en temps constant sur une machine CRCW avec n^2 processeurs.

Exemple: $T[] = 1, 1, 1, 3, 4, 5, 5, 5$, on obtient 1 ou 5 (chacun de ces chiffres a trois occurrences).

2.2* Supposons maintenant que le tableau T ne soit plus limité à des éléments dans $\{1, 2, \dots, n\}$. Ecrire un algorithme sur une machine CRCW avec n processeurs trouvant l'entier qui a le plus d'occurrences dans T en utilisant une machine avec n processeurs.

(Notez que contrairement à la question 2.1 précédente les valeurs dans T ne sont plus bornées et que le nombre de processeurs a diminué).

B. Algorithmique distribuée

Exercice 3. (5 points)

Dans cet exercice, on suppose disposer d'un réseau de n processeurs chacun avec un identifiant unique (un entier positif quelconque) et communiquant avec les $n - 1$ voisins de manière synchrone (chaque processeur connaît donc ses voisins et peut leur envoyer directement des messages sans qu'il y ait collision).

3.1 Ecrire un algorithme déterministe effectuant une élection en rappelant en quoi consiste une élection.

3.2 Quelle est la complexité en temps et quel est le nombre de messages M de votre algorithme?

On reprend l'exercice mais on ne dispose plus d'**identifiant** pour les processeurs.

3.3 Ecrire un algorithme randomisé effectuant l'élection dans l'anneau en supposant que tous les processeurs sont au courant du nombre de participants.

3.4* Quelle est la complexité moyenne en temps et quel est le nombre de messages moyen de votre algorithme?

3.5* Ecrire un algorithme adapté au cas où le nombre de participants n'est pas connu d'avance.

Exercice 4. (5 points)

On dispose d'un réseau complet de n processeurs anonymes communiquant de manière synchrone selon le modèle des "radio networks" avec détecteur de collisions. Chacun de ces processeurs possède une clef (représentée par un entier positif).

4.1 Rappeler les différents modèles.

4.2 Montrer comment on peut calculer la somme des clefs.