



Durée : 2h — Documents autorisés

■ ■ ■ ■ Problème de parallélisation – (8 points)

1 – Une société spécialisée dans les panneaux solaires a mis au point une nouvelle technologie de panneau basée sur l'utilisation de quatre matériaux :

- ★ le gallium, noté dans l'exercice G ;
- ★ le zirconium, noté Z ;
- ★ le thorium, noté T ;
- ★ le cadmium, noté C.

Chaque panneau est construit par un procédé secret de projection qui répartie chaque matériau à la surface du panneau.

Pour vérifier que le panneau est utilisable :

1. on réalise des mesures au microscope électronique qui produisent une matrice de données où la présence de chaque matériau est indiquée par une case contenant sa notation :

T	Z	Z	T	G	C	T	Z	C	G	...
Z	Z	G	T	T	C	C	T	Z	C	...
C	G	Z	C	Z	Z	G	C	T	C	...
C	T	C	G	T	C	C	T	G	Z	...
Z	C	G	G	T	Z	G	C	Z	C	...
T	Z	C	T	C	G	T	Z	T	G	...
T	G	C	Z	C	T	Z	T	Z	G	...
C	Z	T	Z	G	T	C	G	C	C	...
Z	C	C	T	Z	G	G	C	G	T	...
G	T	Z	C	C	T	C	Z	Z	C	...
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

2. on vérifie que la répartition des quatre matériaux sur **la totalité** du panneau est identique, c-à-d qu'il y a bien 25% de chaque matériau ;

On parlera d'étape de vérification I.

3. le panneau est ensuite **découpé**, pour son intégration finale, en **morceaux de taille régulière**.

On vérifie pour **chaque morceau** :

- ◇ qu'il n'y a pas de séquence verticale du même matériau de longueur > 3 ;

On parlera d'étape de vérification II.

- ◇ que la répartition de chaque matériau est encore de 25% ;

On parlera d'étape de vérification III.

Questions

Pour les questions, on considérera un tableau 100x100 **prérempli** de type char, dont chaque case contient un caractère parmi ceux ci : 'C' 'G' 'T' 'Z'.

Ce tableau correspond à la totalité du panneau.

Un morceau correspond au découpage en sous-tableau de 25x25, soient 16 sous-tableaux.

Vous adapterez votre programme à ces valeurs.

- a. Écrire un programme C/OpenMP réalisant l'étape de **vérification I**, c-à-d vérifiant que chaque matériau à un taux de présence de 25% sur le panneau complet.

Votre programme devra indiquer à la fin de son exécution si le panneau est correct ou non.

- b. Écrire un programme C/OpenMP réalisant les étapes de **vérification II & III** en traitant :
 - ◇ chaque matériau séparément ;
 - ◇ par groupe de 4 sous-tableaux (soient 4 fois) ;
- c. Écrire un programme utilisant le plus de *threads* possibles et le plus simple possible en vous inspirant des programmes des questions a) et b).

■ ■ ■ ■ Questions de cours – 6 points

2– Pour la programmation d'un « cluster » à l'aide du modèle « Message Passing », comme proposé par PVM : comment simplifier l'organisation et la mise en place des communications au niveau de l'écriture du programme ?

Vous donnerez un exemple de simplification.

3– Est-ce que la programmation GPGPU proposée par CUDA est plus proche de la directive de parallélisation OpenMP :

- `#pragma omp parallel for`
- `#pragma omp parallel sections`

Vous justifierez votre réponse.

4– Qu'est-ce que l'effet « pipeline » et est-il possible de le mettre en œuvre en OpenMP ?

2pts

■ ■ ■ ■ Programmation OpenMP – 6 points

5– Donnez une version du programme suivant, en utilisant uniquement la directive « `#pragma omp parallel sections` » :

3pts

```
1 #define TAILLE 100
2 ...
3 {
4   int n = TAILLE;
5   int a[TAILLE];
6
7   #pragma omp parallel for schedule(static,25)
8     for (i=0; i<n; i++)
9       a[i] = super_fonction_tres_complexe(i);
10 ...
```

6– Un programme a été décomposé en 5 tâches $\{t_1, t_2, t_3, t_4, t_5\}$.

3pts L'analyse du parallélisme pouvant être exploité entre ces différentes tâches est la suivante :

- ★ les tâches $\{t_1, t_2, t_4, t_5\}$ peuvent être réalisées en parallèle ;
- ★ la tâche t_3 ne peut être faite qu'après les tâches $\{t_1, t_2\}$;
- ★ les tâches t_4 et t_5 ne peuvent être exécutées qu'après la tâche t_3 .

a. Donnez un graphe des dépendances temporelles pour ce programme.

b. Comment à l'aide d'OpenMP, vous pouvez réaliser ce même ordonnancement/exécution parallèle ?
Vous donnerez la structure du programme avec un commentaire pour indiquer le travail de la tâche sous forme de commentaire : `/ Travail ti */`.*