

### Contrôle Final

#### Questions de cours (2Pts X 4=8 Pts)

- Q 1) Donnez une définition brève et concise des termes suivants : -  
- application parallèle, - machine parallèle, - modèle d'architecture parallèle.
- Q 2) Citez les motivations d'utilisation des multi-cœurs par rapport aux multiprocesseurs mono-cœur ?
- Q 3) Expliquez comment le modèle de programmation SPMD peut-~~il~~ <sup>il</sup> emuler le modèle MIMD ?
- Q 4) Les outils de développement d'applications parallèles sont classés selon un seul critère, lequel ?

#### Exercice1 (au choix 6Pts)

- Donnez le code C/MPI de la communication en anneau ( $P_0 \rightarrow P_1 \rightarrow P_2 \dots P_{n-1} \rightarrow P_0$ ) en utilisant la version non bloquante des routines de communication point à point (jeton = valeur entière).
- Portez les corrections nécessaires au code précédent pour traiter le cas de perte du jeton (un processus ne reçoit pas le jeton à cause d'une erreur de communication), le processus en question doit régénérer le jeton (envoi au processus successeur une valeur arbitraire qui représente le nouveau jeton).

#### Exercice2 (au choix 6Pts)

```
public class SharedValue {  
    private int value;  
    public void setV(int val){value=val}  
    public int getV() {return value}  
}
```

- Etant donné deux classes de threads Writer et Reader qui exécutent respectivement setV et getV.
- Modifiez le code précédent pour que le thread Reader n'exécute getV qu'après l'exécution de setV par un thread Writer sur le même objet SharedValue (partagé entre les deux threads).

#### Exercice3 (obligatoire 5Pts)

Soit le code C/MPI suivant :

```
int main(int argc, char ** argv) {  
<PS> /* partie intrinsèquement séquentielle */  
    MPI_Init(&argc, &argv);  
<PP> /* partie parallèle */  
    MPI_Finalize();  
}
```

La routine `double MPI_Wtime()` retourne le temps écoulé à partir d'un moment passé arbitraire unique pour tous les processeurs,

- Modifiez le code précédent pour afficher le temps de début et le temps de fin de la section <PP> en utilisant `MPI_Wtime()` (tous les processus doivent entamer leurs sections <PP> ensemble)
- A l'exécution, chaque processus affichera deux valeurs, quel est le temps d'exécution de la partie parallèle pour l'ensemble des processus ?

**N.B : Choix à faire entre Exercice1 et Exercice2, note de l'Exercice3 à comptabiliser dans la note de TD, 1Pt sur la présentation.**