

UNIVERSITÉ DE SHERBROOKE
DÉPARTEMENT D'INFORMATIQUE

Devoir # 1

IFT 630

Processus concurrents et parallélisme

Chapitre 2 : Concurrence et parallélisme

Question 1

Donnez des similitudes et des différences entre les fils de niveau usager et les coroutines.

Question 2

Soit la suite d'énoncés suivants :

```
1 void calcul1()  
2 {  
3     a = x + y;           // E1  
4     b = z + a;          // E2  
5     c = a - b;          // E3  
6     d = b + 1;          // E4  
7     e = d + 4;          // E5  
8     f = d - 1;          // E6  
9     r = c * (e - f)     // E7  
10 }
```

1. Quels ont les énoncés qui peuvent être exécutés en parallèle? Dessinez le graphe de précedence correspondant. Vous devez faire ressortir le maximum de parallélisme.
2. Écrivez une version parallèle de ce programme en utilisant les énoncés fork/Join (version 1).

3. Écrivez une version parallèle de ce programme en utilisant les énoncés fork/Join (version 2).
4. Écrivez une version parallèle de ce programme en utilisant les énoncés cobegin/coend.

Pour toutes les solutions, utilisez un pseudo-code similaire à celui utilisé dans le cours.

Question 3

Soit la suite d'énoncés suivants :

```
1 void calcul2()
2 {
3     z = 2;           // E0
4     a = x + y;      // E1
5     b = z + a;      // E2
6     r = a ** a;     // E3
7     c = a - 1;      // E4
8     d = z * a;      // E5
9     r = d * r;      // E6
10    printf("",r..); // E7
11 }
```

1. Quels ont les énoncés qui peuvent être exécutés en parallèle? Dessinez le graphe de précedence correspondant. Vous devez faire ressortir le maximum de parallélisme.
2. Écrivez une version parallèle de ce programme en utilisant les énoncés fork/Join (version 1).
3. Écrivez une version parallèle de ce programme en utilisant les énoncés fork/Join (version 2).
4. Écrivez une version parallèle de ce programme en utilisant les énoncés cobegin/coend.

Pour toutes les solutions, utilisez un pseudo-code similaire à celui utilisé dans le cours.