

Série d'exercices #2

IFT-2245

January 22, 2017

2.1 Systèmes distribués

Expliquer la différence entre le modèle client-serveur et le modèle décentralisé *peer-to-peer* de systèmes distribués.

2.2 Programme vs processus

Expliquer ce qui distingue un programme (*program*) d'un processus (*process*)? Indice: Penser à la relation entre les deux et où ils résident.

2.3 Ordonnanceur

Donner une différence entre ordonnanceur à court terme et à long terme (*long-term* and *short-term* schedulers).

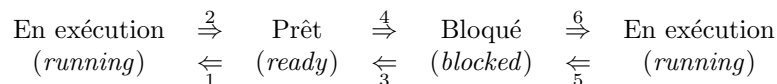
2.4 Création de *thread*

Quand un *thread* crée un nouveau *thread*, quelles parties, parmi les suivantes, sont-elles partagées entre les deux *threads*:

- Les registres
- Le tas
- Les variables globales
- La pile

2.5 L'état d'un *thread*

Pour chaque flèche dans le diagramme ci-bas, décrire une condition qui causerait un fil (thread) de passer d'un état à l'autre. Indiquer "impossible" si cela ne peut pas se produire. S'il y a lieu, indiquer aussi quelle action d'un *context switch* s'y produit (save ou restore).



2.6 Context switch

Décrire les actions entreprises par le noyau lorsqu'il change d'un processus à un autre (*context switch*).

2.7 Fork

Quand un processus crée un nouveau processus avec l'opération `fork`, quelles parties, parmi les suivantes, sont-elles partagées entre le processus parent et le processus enfant:

- La pile
- Le tas
- Les segments de mémoire partagés

2.8 Fork²

Soit le code suivant:

```
#include <stdio.h>
int value = 5;
int main () {
    if (fork () == 0)
        value += 15;
    else {
        wait (NULL);
        printf ("Value = %d\n", value);
    }
    return 0;
}
```

Quelle valeur va-t-il renvoyer à l'écran?

2.9 Fork³

Soit le code suivant:

```
int main () {
    fork (); fork (); fork (); return 1;
}
```

Incluant le processus initial, combien de processus sont créés par ce programme?