le 20 janvier 2003

2 Heures
Documents autorisés :
notes manuscrites et
polycopié de cours

Système

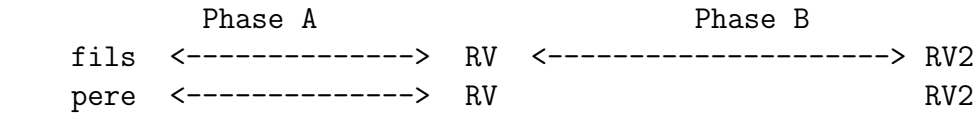
Licence Informatique

Exercice 1 Quel est la signification du terme attente active ? Quels autres types d'attente existent-ils ? Comment peut-on éviter l'attente active ?

Exercice 2 La plupart des ordonnanceurs circulaires utilisent un quantum fixe. Donnez un argument en faveur d'un petit quantum et un autre en faveur d'un grand quantum .

Exercice 3 Dans un système de file de messages, on dispose des opérations atomiques suivantes : `creerfile(cle) --> cle_interne`
`ecrirefile(cle_interne, message)`
`lirefile(cle_interne, adresse_message)`
L'operation `ecrirefile` (resp. `lirefile`) est bloquante si la file est pleine (resp. vide). Réécrire les opérations P, V et Initsem des semaphores au moyen des opérations sur les files de messages décrites ci-dessus.

Exercice 4 On souhaite réaliser la synchronisation de deux processus père et fils au moyen d'un sémaphore unique, les deux processus correspondant à l'itération infinie du schéma suivant :



L'execution des deux processus doit donc etre telle que :

1. Le fils ne peut entrer dans sa phase B qu'après que son père a d'une part terminé sa phase A et d'autre part pris connaissance que lui même a terminé sa phase A (RV)
2. Le processus père ne reprend sa phase A que quand le processus fils a terminé sa phase B et va rentrer lui même en phase A (RV2)
3. Les durées des phases A des deux processus et de la phase B du fils sont variables.

Ecrire l'algorithme du père et du fils.

Exercice 5 Décrire l'arbre des processus ainsi que les operations effectuées par chacun de ceux-ci correspondant au lancement du programme suivant.

```
main(){
if (fork() == 0)
    if (fork() == 0)
        sleep(10);
    else {
        execlp("a","a",NULL);
        exit(2);}
if (fork() == 0) {
    execlp("b","b",NULL);
    exit(2);}
if (fork() == 0) {
    execlp("c","c",NULL);
    exit(2);}
```

$\alpha()$

```
pid1=wait(NULL);
sleep(1);
pid2=wait(NULL);
sleep(3);
```

$\alpha()$

```
pid3=wait(NULL);
pid4=wait(NULL);}
```

On suppose que l'ensemble des variables est défini ailleurs. α est une fonction définie. a, b, c sont des programmes exécutables.