

Pr FARAH
Département d'informatique
Promotion Licence 3eme année
Documents non autorisés
Module : SE2

Le 26/01/2023

Durée 1h30

EMD1

Exercice 1 : (10 pts)

Supposons 3 processus P1, P2, P3. P1 produit en premier un message qu'il déposera dans une case C1, le processus P2 prend de C1 traite le message et dépose un autre message dans une case C2. P3 prend le message de C2 et le consomme.

Donner les programmes des trois processus en utilisant les sémaphores.

Exercice 2 : (10 pts)

Dans une bibliothèque on a des clients qui arrivent, cette bibliothèque met à disposition des sacs au nombre de N, s'il n'y a plus de sac le client ne peut pas entrer. Deux types de clients existent, les abonnés et les non abonnés. On veut synchroniser les processus de ce système, en sachant qu'il n'y a pas de problème d'exclusion mutuelle entre abonnés et non abonnés.

Ecrire les algorithmes des processus « abonnés » et « non abonnés ».

Pr FARAH
Département d'informatique
Promotion Licence 3eme année
Documents non autorisés
Module : SE2

Le 26/01/2023

Durée 1h30

EMD1

Exercice 1 : (10 pts)

Supposons 3 processus P1, P2, P3. P1 produit en premier un message qu'il déposera dans une case C1, le processus P2 prend de C1 traite le message et dépose un autre message dans une case C2. P3 prend le message de C2 et le consomme.

Donner les programmes des trois processus en utilisant les sémaphores.

Exercice 2 : (10 pts)

Dans une bibliothèque on a des clients qui arrivent, cette bibliothèque met à disposition des sacs au nombre de N, s'il n'y a plus de sac le client ne peut pas entrer. Deux types de clients existent, les abonnés et les non abonnés. On veut synchroniser les processus de ce système, en sachant qu'il n'y a pas de problème d'exclusion mutuelle entre abonnés et non abonnés.

Ecrire les algorithmes des processus « abonnés » et « non abonnés ».

Exercice 1

```
var S1,S2,S3,S4,mutex1,mutex2 : sémaphore init n1,0,n2,0,1,1 ;  
/* ni=taille de Ci, i=1,2 */
```

Processus P1

Début

```
A1 : <Produire un message>  
P(S1)  
P(mutex1)  
<Dépôt du message dans C1>  
V(mutex1)  
V(S2)  
Aller_à A1  
Fin
```

Processus P2

Début

```
A2 : P(S2)  
P(mutex1)  
<Prélever un message de C1>  
V(mutex1)  
V(S1)  
<Traiter le message>  
P(S3)  
P(mutex2)  
<Dépôt du résultat dans C2>  
V(mutex2)  
V(S4)  
Aller_à A2  
Fin
```

Processus P3

Début

```
A3 : P(S4)  
P(mutex2)  
<Prélever un message de C2>  
V(mutex2)  
V(S3)  
<Traitement du message>  
Aller_à A3  
Fin
```

Exercice 2 :

```
var S1,S2,mutex : sémaphore init 0,0,1 ; /* INIT S1.val=0; S2.val=0;  
mutex.val=1 */
```

```
NbC, na1, na2 : entier init 0,0,0 ; /* NbC=0; na1=0; na2=0 */
```

Les algorithmes des processus en question seront exprimés comme suit :

Processus Client_Abonné

Début

```
P(mutex)  
si NbC < N alors  
NbC := NbC + 1  
V(mutex)  
sinon  
na1 := na1 + 1  
V(mutex)  
P(S1)
```

finsi

Processus Client_Non_Abonné

Début

```
P(mutex)  
si NbC < N alors  
NbC := NbC + 1  
V(mutex)  
sinon  
na2 := na2 + 1  
V(mutex)  
P(S2)
```

finsi

```
Effectuer les achats
P(mutex)
si na1 > 0 alors
    na1 := na1 - 1
    V(S1)
sinon /* na1=0 */
    si na2 > 0 alors
        na2 := na2 - 1
        V(S2)
    sinon /* na1=na2=0 */
        NbC := NbC - 1
    finsi
finsi
V(mutex)
Fin
```

```
Effectuer les achats
P(mutex)
si na1>0 alors
    na1 := na1 - 1
    V(S1)
sinon /* na1=0 */
    si na2 > 0 alors
        na2 := na2 - 1
        V(S2)
    sinon /* na1=na2=0 */
        NbC := NbC - 1
    finsi
finsi
V(mutex)
```

Fin