

Compte rendu tp3 système

Dans ce TP nous allons utiliser des sémaphores pour implanter un modèle producteur/consommateur similaire à celui que nous avons vu en TD.

Question 1 : on corrige les erreurs!

Question 2 :



On initialise `sem_1 = 0` et `sem_2 = 0`.

L'algorithme précédent est autobloquant car le processus 1 attend `sem_2` et le processus 2 attend `sem_1`.

Voici la solution que je propose:

```

void *p1(void *arg)
{
    a1();
    /* send signal to the created thread */

    sem_post(sync_1);
    sem_wait(sync_2);
    a2();
}

void *p2(void *arg)
{
    sem_wait(sync_1);
    b1();
    /* wait for signal from main thread */

    sem_post(sync_2);
    b2();
}

```

Question 3:

```
init(&mutex, 1);
```

Thread A

```

wait(&mutex);
/* CS */
count = count + 1;
signal(&mutex);

```

Thread B

```

wait(&mutex);
/* CS */
count = count + 1;
signal(&mutex);

```

Ce pseudo code garantie l'exclusion mutuelle car le mutex est initialisé à 1, ainsi il ne peut y avoir que un seul thread de lancé à la fois

Voici le résultat de ex2 corriger:

```
x:1, i:1
x:2, i:2
x:3, i:4
x:4, i:5
x:5, i:5
x:6, i:6
x:7, i:8
x:8, i:9
x:9, i:9
x:10, i:4
Final value of x is 10
adam@DESKTOP-CCBFT0R: /m
```

Question 4:

On implémente un premier modèle de producteur consommateur où il n'y a qu'un seul producteur et qu'un seul consommateur.

La fonction retirer permet de copier la variable_global dans une variable en parametre.

La fonction déposer permet de copier la variable en parametre dans la variable global.

Mais pour le moment ces fonctions ne sont pas utile au fonctionnement du programme.

Notre programme fonctionne en voici le résultat:

```
MESSAGE : 54232  
MESSAGE : 54233  
MESSAGE : 54234  
MESSAGE : 54235  
MESSAGE : 54236  
MESSAGE : 54237  
MESSAGE : 54238  
MESSAGE : 54239  
MESSAGE : 54240  
MESSAGE : 54241  
MESSAGE : 54242  
MESSAGE : 54243  
MESSAGE : 54244  
MESSAGE : 54245  
MESSAGE : 54246  
MESSAGE : 54247  
MESSAGE : 54248  
MESSAGE : 54249  
MESSAGE : 54250  
MESSAGE : 54251  
MESSAGE : 54252  
MESSAGE : 54253  
MESSAGE : 54254  
MESSAGE : 54255  
MESSAGE : 54256
```

Question 5: