

Compte rendu tp System d'exploitation

master 1 Rsd

Etudiant: Bouzara zakaria
Matricule: 212138069681

Exercice 3:

1. les codes attachés:

```
exo3 /  
- semaphore.h  
- spool.c
```

2. Résultat d'exécution:

```
SE/cr/exo3 via C v14.2.1-gcc  
❯ ./spool  
New semaphore group created with id (2)  
Init semctl successful  
Enter the number of child processes: 9  
Child 0 with PID (77195): Waiting  
P semop executed successfully  
Child 0 with PID (77195): Executing  
Child 1 with PID (77196): Waiting  
Child 2 with PID (77197): Waiting  
Child 3 with PID (77198): Waiting  
Child 4 with PID (77199): Waiting  
Child 5 with PID (77200): Waiting  
Child 6 with PID (77201): Waiting  
Child 7 with PID (77202): Waiting  
Child 8 with PID (77203): Waiting  
  
Child 0 with PID (77195): Completed in 1 seconds  
V semop executed successfully  
P semop executed successfully  
Child 1 with PID (77196): Executing  
P semop executed successfully  
Child 2 with PID (77197): Executing  
Child 1 with PID (77196): Completed in 1 seconds  
V semop executed successfully  
Child 2 with PID (77197): Completed in 1 seconds  
P semop executed successfully  
Child 3 with PID (77198): Executing  
V semop executed successfully  
P semop executed successfully  
Child 5 with PID (77200): Executing  
P semop executed successfully  
Child 4 with PID (77199): Executing  
P semop executed successfully  
Child 6 with PID (77201): Executing  
Child 3 with PID (77198): Completed in 1 seconds  
V semop executed successfully  
Child 4 with PID (77199): Completed in 1 seconds  
V semop executed successfully  
Child 5 with PID (77200): Completed in 1 seconds  
V semop executed successfully  
P semop executed successfully  
Child 8 with PID (77203): Executing  
Child 6 with PID (77201): Completed in 1 seconds  
V semop executed successfully  
P semop executed successfully  
Child 7 with PID (77202): Executing  
Child 8 with PID (77203): Completed in 1 seconds  
V semop executed successfully  
Child 7 with PID (77202): Completed in 1 seconds  
V semop executed successfully  
Parent process with PID (77194): All children have finished  
Semaphore set removed successfully.
```

Si un processus se termine de manière inattendue (par exemple, en recevant le signal **SIGKILL** via la commande `kill -9 <pid>`) alors qu'il utilise une ressource partagée, la valeur du sémaphore peut rester décrémentée, bloquant ainsi l'accès à la ressource indéfiniment. Cela se produit parce que l'opération P (appel à `semop` avec `sem_op = -1`) décrémente le sémaphore, mais l'opération V correspondante (appel à `semop` avec `sem_op = 1`) n'est jamais exécutée en raison de l'arrêt prématuré du processus. Pour remédier à cette situation, utilisez le drapeau `SEM_UNDO` lors de l'opération P(`semop`).

Ce drapeau assure que toute décrémentation effectuée sur le sémaphore par un processus est automatiquement annulée par le système d'exploitation si ce processus se termine de manière inattendue.

Exercice 4:

1. les codes attachés:

```
exo4 /  
- Create.c  
- Consumer.c  
- Producer.c  
- semaphore.h
```

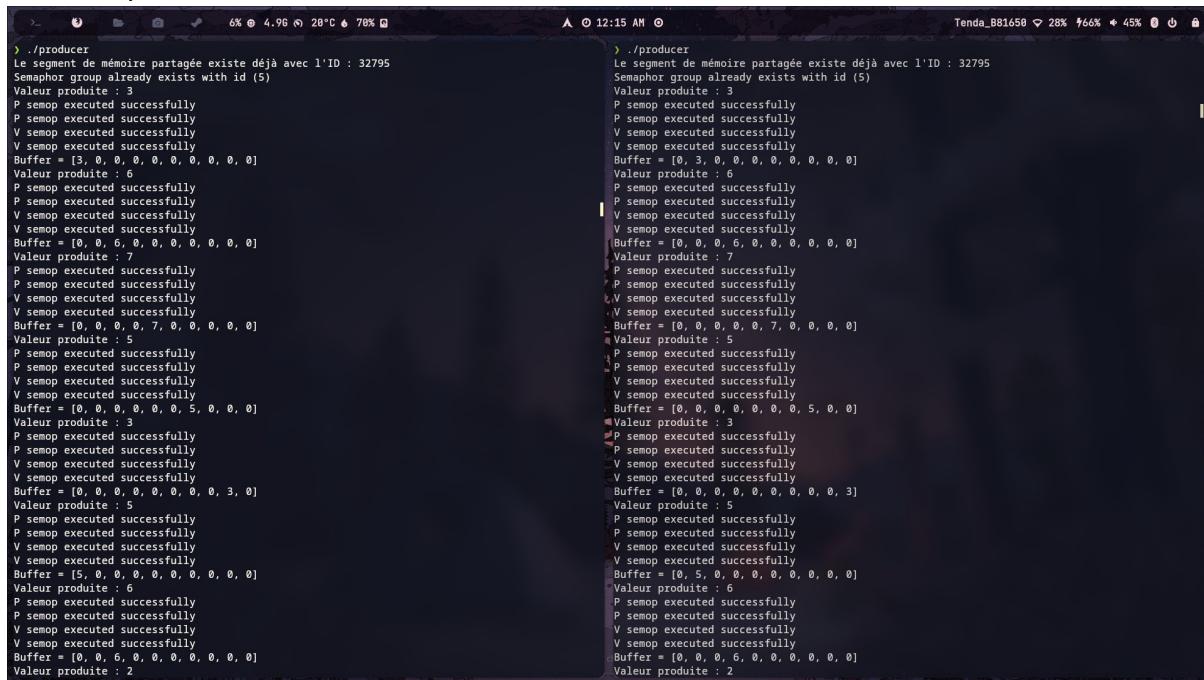
2. Résultat d'exécution:

Create.c:

```
SE/cr/exo4 via C v14.2.1-gcc took 3m20s  
> gcc -o create create.c  
  
SE/cr/exo4 via C v14.2.1-gcc  
> ./create  
Nouveau segment de mémoire partagée créé avec l'ID : 32795  
Indexes initialisés (Producteur : 0, Consommateur : 0)  
New semaphore group created with id (5)  
Init semctl successful  
Init semctl successful  
Init semctl successful  
Init semctl successful
```

L'exécution en parallèle:

2 producteurs

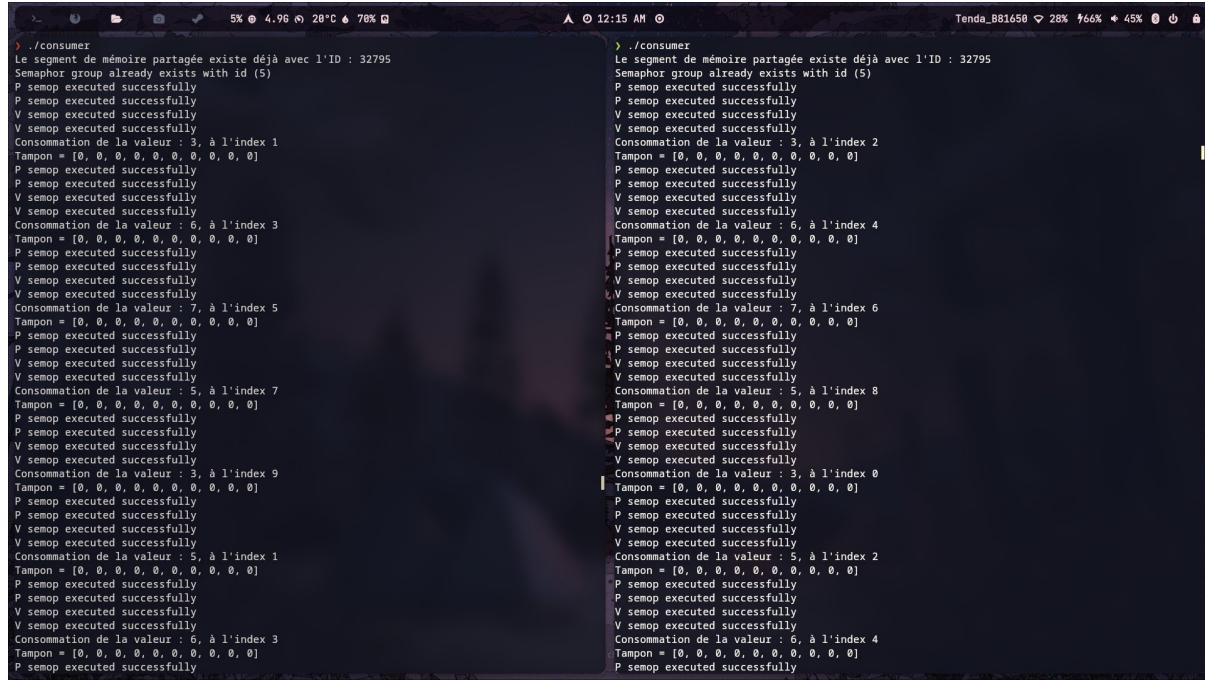


The screenshot shows two terminal windows side-by-side. Both windows have a black background and white text. The left window title is 'A 6% 4.96 28°C 78%' and the right window title is 'A 12:15 AM Tenda_B8165B 28% 766% 65%'. Both windows show the command 'gcc -o producer producer.c' being run. The output of both windows is identical, showing the execution of semaphores and the modification of a shared buffer. The buffer starts at [3, 0, 0, 0, 0, 0, 0, 0, 0] and ends at [0, 0, 0, 0, 0, 0, 0, 0, 0]. The product value increases from 3 to 6 to 9.

```
./producer  
Le segment de mémoire partagée existe déjà avec l'ID : 32795  
Semaphore group already exists with id (5)  
Valeur produite : 3  
P semop executed successfully  
P semop executed successfully  
V semop executed successfully  
V semop executed successfully  
Buffer = [3, 0, 0, 0, 0, 0, 0, 0, 0]  
Valeur produite : 6  
P semop executed successfully  
P semop executed successfully  
V semop executed successfully  
V semop executed successfully  
Buffer = [0, 0, 6, 0, 0, 0, 0, 0, 0]  
Valeur produite : 7  
P semop executed successfully  
P semop executed successfully  
V semop executed successfully  
V semop executed successfully  
Buffer = [0, 0, 0, 0, 7, 0, 0, 0, 0]  
Valeur produite : 5  
P semop executed successfully  
P semop executed successfully  
V semop executed successfully  
V semop executed successfully  
Buffer = [0, 0, 0, 0, 0, 0, 5, 0, 0]  
Valeur produite : 3  
P semop executed successfully  
P semop executed successfully  
V semop executed successfully  
V semop executed successfully  
Buffer = [0, 0, 0, 0, 0, 0, 0, 5, 0]  
Valeur produite : 6  
P semop executed successfully  
P semop executed successfully  
V semop executed successfully  
V semop executed successfully  
Buffer = [0, 0, 0, 0, 0, 0, 0, 0, 6]  
Valeur produite : 9  
P semop executed successfully  
P semop executed successfully  
V semop executed successfully  
V semop executed successfully  
Buffer = [0, 0, 0, 0, 0, 0, 0, 0, 0]
```

```
./producer  
Le segment de mémoire partagée existe déjà avec l'ID : 32795  
Semaphore group already exists with id (5)  
Valeur produite : 3  
P semop executed successfully  
P semop executed successfully  
V semop executed successfully  
V semop executed successfully  
Buffer = [0, 0, 0, 0, 0, 0, 0, 0, 0]  
Valeur produite : 6  
P semop executed successfully  
P semop executed successfully  
V semop executed successfully  
V semop executed successfully  
Buffer = [0, 0, 0, 0, 0, 0, 7, 0, 0]  
Valeur produite : 9  
P semop executed successfully  
P semop executed successfully  
V semop executed successfully  
V semop executed successfully  
Buffer = [0, 0, 0, 0, 0, 0, 0, 5, 0]  
Valeur produite : 3  
P semop executed successfully  
P semop executed successfully  
V semop executed successfully  
V semop executed successfully  
Buffer = [0, 0, 0, 0, 0, 0, 0, 0, 6]  
Valeur produite : 6  
P semop executed successfully  
P semop executed successfully  
V semop executed successfully  
V semop executed successfully  
Buffer = [0, 0, 0, 0, 0, 0, 0, 0, 0]
```

2 Consommateurs:



```
> ./consumer
Le segment de mémoire partagée existe déjà avec l'ID : 32795
Semaphore group already exists with id (5)
P semop executed successfully
P semop executed successfully
V semop executed successfully
V semop executed successfully
Consummation de la valeur : 3, à l'index 1
Tampon = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
P semop executed successfully
P semop executed successfully
V semop executed successfully
V semop executed successfully
V semop executed successfully
Consummation de la valeur : 3, à l'index 2
Tampon = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
P semop executed successfully
P semop executed successfully
V semop executed successfully
V semop executed successfully
V semop executed successfully
Consummation de la valeur : 3, à l'index 3
Tampon = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
P semop executed successfully
P semop executed successfully
V semop executed successfully
V semop executed successfully
V semop executed successfully
Consummation de la valeur : 3, à l'index 4
Tampon = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
P semop executed successfully
P semop executed successfully
V semop executed successfully
V semop executed successfully
V semop executed successfully
Consummation de la valeur : 3, à l'index 5
Tampon = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
P semop executed successfully
P semop executed successfully
V semop executed successfully
V semop executed successfully
V semop executed successfully
Consummation de la valeur : 3, à l'index 6
Tampon = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
P semop executed successfully
P semop executed successfully
V semop executed successfully
V semop executed successfully
V semop executed successfully
Consummation de la valeur : 3, à l'index 7
Tampon = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
P semop executed successfully
P semop executed successfully
V semop executed successfully
V semop executed successfully
V semop executed successfully
Consummation de la valeur : 3, à l'index 8
Tampon = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
P semop executed successfully
P semop executed successfully
V semop executed successfully
V semop executed successfully
V semop executed successfully
Consummation de la valeur : 3, à l'index 0
Tampon = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
P semop executed successfully
P semop executed successfully
V semop executed successfully
V semop executed successfully
V semop executed successfully
Consummation de la valeur : 3, à l'index 1
Tampon = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
P semop executed successfully
P semop executed successfully
V semop executed successfully
V semop executed successfully
V semop executed successfully
Consummation de la valeur : 3, à l'index 2
Tampon = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
P semop executed successfully
P semop executed successfully
V semop executed successfully
V semop executed successfully
V semop executed successfully
Consummation de la valeur : 3, à l'index 3
Tampon = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
P semop executed successfully
P semop executed successfully
V semop executed successfully
V semop executed successfully
V semop executed successfully
Consummation de la valeur : 3, à l'index 4
Tampon = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
P semop executed successfully
P semop executed successfully
V semop executed successfully
V semop executed successfully
V semop executed successfully
```

Si le producteur est arrêté :

Le consommateur pourra toujours accéder à la mémoire partagée et vérifier le tampon pour consommer des valeurs. Cependant, comme le producteur ne produit plus de nouvelles valeurs, le consommateur trouvera constamment le tampon vide (les valeurs resteront à 0). Le sémaphore `empty_slots` (sémaphore 0) restera inchangé car le producteur ne met pas à jour le tampon. Ce sémaphore devrait être dans un état où le consommateur attend que le sémaphore `empty_slots` soit signalé par le producteur. Mais comme aucune valeur n'est produite, le consommateur attendra indéfiniment.

Si le consommateur est arrêté :

Le producteur continuera à produire des valeurs et à les placer dans le tampon, en mettant à jour tab et en signalant le sémaphore `full_slots` (sémaphore 1) pour indiquer que des données sont disponibles pour la consommation. Cependant, comme le consommateur ne consomme pas les éléments, le sémaphore `full_slots` continuera à augmenter, mais le sémaphore `empty_slots` ne sera pas libéré. Cela peut amener le producteur à se bloquer ou à ne pas pouvoir insérer de nouvelles valeurs une fois que le tampon est plein, car le producteur attend de l'espace dans le tampon (`empty_slots`).

Exercice 4:

les codes attachés:

```
exo5 /  
  - Create.c  
  - Oxygen.c  
  - semaphore.h
```

1. si on arrête un des processus :

Si vous arrêtez un des processus ([hydrogen ou oxygen](#)), l'autre processus continuera à produire des valeurs jusqu'à ce que le tampon soit plein, puis il se bloquera en attendant que le consommateur consomme des valeurs.

À l'inverse, si vous arrêtez le producteur ([oxygen](#)), le consommateur ([hydrogen](#)) se bloquera en attendant que de nouvelles valeurs soient produites une fois qu'il aura consommé toutes les valeurs actuelles dans le tampon.

```
./hydrogen  
Semaphore group already exists with id (0)  
Memory segment already exists with ID (10)  
P semop executed successfully  
V semop executed successfully  
P semop executed successfully  
P semop executed successfully  
valeur du compteur : 1  
V semop executed successfully  
P semop executed successfully  
P semop executed successfully  
P semop executed successfully  
H2O formed with Oxy PID: 5573  
Blocked at barrier waiting for 1Hydrogen or 10xygen...  
P semop executed successfully  
H2O formed with Hydro PID: 5605  
SE/cr/exo5 via C v14.2.1-gcc  
> ipcs -s  
----- Semaphore Arrays -----  
key      semid      owner      perms      nsems  
0x6108ff2f  0        Zakkye    666       4  
SE/cr/exo5 via C v14.2.1-gcc  
> ipcs -si 0  
Semaphore Array semid=0  
uid=1000      gid=1000      cuid=1000      cgid=1000  
mode=0666, access_perms=0666  
nsems = 4  
otime = Fri Dec 27 19:46:33 2024  
ctime = Fri Dec 27 19:10:04 2024  
semnum   value      ncount      zcount      pid  
0        1          0          0          5637  
1        2          0          0          5637  
2        1          0          0          5637  
3        2          0          0          5637
```