

[Tableau de bord](#) / [Mes cours](#) / [INF2610 - Noyau d'un système d'exploitation](#) / Semaine 7 - Contrôle périodique (9h30-12h00)  
/ [Contrôle Périodique](#)

**Commencé le** mercredi 21 octobre 2020, 09:28

**État** Terminé

**Terminé le** mercredi 21 octobre 2020, 12:00

**Temps mis** 2 heures 31 min

**Note** 34,73 sur 50,00 (69%)

Question 1

Terminer

Non noté

J'affirme que je compléterai cet examen en vertu du code de conduite de l'étudiant de Polytechnique Montréal et de sa politique sur le plagiat.

J'affirme que j'utiliserai seulement le matériel autorisé tel que décrit dans le plan de cours.

Sur mon honneur, j'affirme que je compléterai cet examen par moi-même, sans communication avec personne, et selon les directives diffusées sur les canaux de communication.

Écrivez votre **nom complet** ainsi que votre **matricule** en guise d'approbation dans la zone de texte ci-dessous.

Victor Kim 1954607

Question 2

Correct

Note de 1,00  
sur 1,00

Un défaut de page génère une interruption matérielle.

Sélectionnez une réponse :

☐ Vrai

☒ Faux ✓

La réponse correcte est « Faux ».

Question 3

Correct

Note de 1,00  
sur 1,00

Pour signaler la fin d'un certain délai à l'ordonnanceur, une interruption logicielle sera générée.

Sélectionnez une réponse :

☐ Vrai

- ☐ Vrai
- ☒ Faux ✓

La réponse correcte est « Faux ».

**Question 4**

Incorrect

Note de 0,00  
sur 1,00

L'exécution d'un processus doit progresser séquentiellement, c'est-à-dire, à n'importe quel moment une seule instruction au plus est exécutée au nom du processus.

Sélectionnez une réponse :

- ☐ Vrai
- ☒ Faux ✗

La réponse correcte est « Vrai ».

**Question 5**

Correct

Note de 1,00  
sur 1,00

Les algorithmes d'ordonnancement préemptif sélectionnent un processus et le laisse s'exécuter pendant un quantum de temps.

Sélectionnez une réponse :

- ☒ Vrai ✓
- ☐ Faux

La réponse correcte est « Vrai ».

**Question 6**

Correct

Note de 1,00  
sur 1,00

Dans Linux, l'appel système SIGKILL permet à un processus de forcer la terminaison d'un autre processus.

Sélectionnez une réponse :

- ☒ Vrai ✓
- ☐ Faux

La réponse correcte est « Vrai ».

Commentaire :

**Question 7**

Correct

Un seul processus peut être en exécution sur n'importe quel processeur à tout moment.



Correct

Note de 1,00  
sur 1,00

Sélectionnez une réponse :

- ☒ Vrai ✓
- ☐ Faux

La réponse correcte est « Vrai ».

Question 8

Incorrect

Note de 0,00  
sur 1,00

Les fils d'exécution améliorent la réactivité.

Sélectionnez une réponse :

- ☐ Vrai
- ☒ Faux ✗

La réponse correcte est « Vrai ».

Question 9

Correct

Note de 1,00  
sur 1,00

Pour accéder à un tube nommé, un processus doit d'abord l'ouvrir en écriture.

Sélectionnez une réponse :

- ☐ Vrai
- ☒ Faux ✓

La réponse correcte est « Faux ».

Question 10

Correct

Note de 1,00  
sur 1,00

À la différence d'un tube anonyme, l'ouverture d'un tube nommé est bloquante.

Sélectionnez une réponse :

- ☒ Vrai ✓
- ☐ Faux

La réponse correcte est « Vrai ».

Question 11

Correct

Les informations stockées dans un tube de communication sont consommées par les processus écrivains selon une politique FIFO.

Note de 1,00  
sur 1,00

Sélectionnez une réponse :

- ☐ Vrai
- ☒ Faux ✓

La réponse correcte est « Faux ».

Question 12

Incorrect

Note de 0,00  
sur 1,00

Un tube de communication permet une communication unidirectionnelle entre des fils d'exécution.

Sélectionnez une réponse :

- ☒ Vrai ✗
- ☐ Faux

Entre des processus

La réponse correcte est « Faux ».

Question 13

Correct

Note de 1,00  
sur 1,00

La table de bits des signaux en attente masque des signaux du processus.

Sélectionnez une réponse :

- ☐ Vrai
- ☒ Faux ✓

La réponse correcte est « Faux ».

Question 14

Correct

Note de 1,00  
sur 1,00

L'utilisation de sémaphore est une solution à l'attente active.

Sélectionnez une réponse :

- ☒ Vrai ✓
- ☐ Faux

La réponse correcte est « Vrai ».

Question 15

Dans le problème du producteur-consommateur, un tampon plein bloque le consommateur.

Correct

Note de 1,00  
sur 1,00

Dans le problème du producteur-consommateur, un tampon plein bloque le consommateur.

Sélectionnez une réponse :

- ☐ Vrai
- ☒ Faux ✓

La réponse correcte est « Faux ».

Question 16

Correct

Note de 1,00  
sur 1,00

L'attente active d'un verrou peut être réalisée par une boucle while sur une fonction atomique.

Sélectionnez une réponse :

- ☒ Vrai ✓
- ☐ Faux

La réponse correcte est « Vrai ».

Question 17

Correct

Note de 1,00  
sur 1,00

Il faut absolument éviter d'utiliser les mécanismes de synchronisation comme les sémaphores ou les mutex dans les routines de gestion d'interruption.

Sélectionnez une réponse :

- ☒ Vrai ✓
- ☐ Faux

La réponse correcte est « Vrai ».

Question 18

Partiellement  
correctNote de 0,60  
sur 1,00

Plusieurs architectures de systèmes d'exploitation existent, dont :

- ☒ Hybride
- ☐ Client-serveur
- ☒ Monolithique
- ☐ Machine virtuelle
- ☒ Micro-noyau

✓

✓

✓

Votre réponse est partiellement correcte.

Vous en avez sélectionné correctement 3.

Les réponses correctes sont :

Monolithique,

Client-serveur,

Micro-noyau,

Hybride,

Machine virtuelle

Question **19**

Terminer

Non noté

Justifiez votre choix de réponse (Si votre justification est absente ou erronée vous aurez **0 point** pour cette question que votre réponse dans la QCM soit bonne ou mauvaise)

Le modèle noyau monolithique est le modèle où un ensemble de modules peuvent être dynamiquement chargés en mémoire physique et exécutés en mode noyau. Le modèle Micro-noyau laisse la majorité des fonctionnalités en mode utilisateur et un minimum de fonctionnalités essentielles en mode noyau ce modèle permet de modifier et ajouter des services plus facilement. Pour le modèle noyau hybride, c'est un mélange de noyau monolithique et du micro-noyau, il permet de charger les services dynamiquement en mémoire physique et la majorité des fonctionnalités sont en mode utilisateur et une petite partie s'exécute en mode noyau.

Question **20**

Correct

Note de 1,00  
sur 1,00

Appel système du noyau Linux qui crée un processus fils selon le principe de *Copy-On-Write* :

- ☐ Pthread\_create()
- ☐ Pipe()
- ☐ Pth\_spawn()
- ☒ Fork()



☐ Aucune de ces réponses

Votre réponse est correcte.

La réponse correcte est :

Fork()

Question **21**

Terminer

Non noté

Justifiez votre choix de réponse (Si votre justification est absente ou erronée vous aurez **0 point** pour cette question que votre réponse dans la QCM soit bonne ou mauvaise)

l'appel système fork() va créer un processus fils qui est un clone de son père. Sa création est fait selon le principe de copy on write donc la lecture d'une attribut de son père, le fils va aller lire directement la donnée de son père et pour un écriture, le fils va copier la donnée de son père. le fils a le même code que celui de son père et il va exécuter les instructions qui suivent l'appel à fork().

Question **22**

Correct

Note de 1,00  
sur 1,00

Quelle est la valeur de retour de la fonction waitpid() si le processus est terminé :

☐ Aucune de ces réponses

☒ PID



☐ 0☐ PPID☐ -1

Votre réponse est correcte.

La réponse correcte est :  
PID

Question **23**

Terminer

Non noté

Justifiez votre choix de réponse (Si votre justification est absente ou erronée vous aurez **0 point** pour cette question que votre réponse dans la QCM soit bonne ou mauvaise)

Le processus père qui exécute waitpid() va attendre le processus fils et quand le processus fils termine avec succès, l'appel système waitpid() va retourner le pid du fils terminé. Sinon s'il y a un échec, il va retourner -1.


Question **24**

Correct

Dans quelle(s) condition(s) un processus passera de l'état Prêt à Élu :



Note de 1,00  
sur 1,00

- ☐ Le processus attend la réponse d'un périphérique
- ☐ Le processus accède à un périphérique ou doit attendre
- ☐ Le périphérique a répondu au processus ou l'attente est finie
- ☐ Le système d'exploitation décide d'exécuter un autre processus
- ☒ Le système d'exploitation décide d'exécuter le processus 

Votre réponse est correcte.

La réponse correcte est :

Le système d'exploitation décide d'exécuter le processus

Question **25**

Terminer

Non noté

Justifiez votre choix de réponse (Si votre justification est absente ou erronée vous aurez **0 point** pour cette question que votre réponse dans la QCM soit bonne ou mauvaise)

Pour qu'un processus passe de l'état prêt à l'état élu, s'il est déjà en train de s'exécuter par le microprocesseur donc quand il aura obtenu la ressource après sa demande. Une seul processus peut passer à l'état élu à chaque instant.

Question **26**

Incorrect

Note de 0,00  
sur 1,00

Le bout de code suivant est-il déterministe?

```
int a = 0;
void *increment(void *)
{
    a++;
    return NULL;
}
int main()
{
    pthread_t th[3];
    int i;
    for (i = 0; i < 3; i++)
        pthread_create(&th[i], NULL, increment, NULL);
    printf(" a = %d\n", a);
    return 0;
}
```

- ☐ Aucune de ces réponses
- ☐ Non, car il affichera 0, 1, 2 ou 3
- ☐ Oui, car il affichera toujours la valeur 3
- ☒ Non, car il affichera 1, 2 ou 3
- ☐ Oui, car il affichera toujours 0



Votre réponse est incorrecte.

La réponse correcte est :

Non, car il affichera 0, 1, 2 ou 3

Question **27**

Terminer

Non noté

Justifiez votre choix de réponse (Si votre justification est absente ou erronée vous aurez **0 point** pour cette question que votre réponse dans la QCM soit bonne ou mauvaise)

Non ce n'ai pas déterministe car chaque thread créer va print un valeur différente de la variable globale a. Les ressources des threads sont partagés donc la modification va être vu par les autres.

Question **28**

Incorrect

Note de 0,00  
sur 1,00

Le système d'exploitation envoie le signal SIGCHLD au processus lorsque :

- ☐ Le processus tente d'écrire dans un tube de communication rompu
- ☐ Le processus se termine
- ☐ Le tube de communication n'a pas de lecteur
- ☐ Le processus tente de lire dans un tube de communication vide
- ☒ Aucune de ces réponses



Votre réponse est incorrecte.

La réponse correcte est :

Le processus se termine

Question **29**

Terminer

Non noté

Justifiez votre choix de réponse (Si votre justification est absente ou erronée vous aurez **0 point** pour cette question que votre réponse dans la QCM soit bonne ou mauvaise)

Le système d'exploitation envoie SIGCHLD au processus père quand le processus fils s'arrête ou se termine.

Question **30**

Correct

Note de 1,00  
sur 1,00

Pour accéder à un tube nommé, un processus doit d'abord l'ouvrir, dans quelle(s) situation(s) un processus passe à l'état Bloqué

- ☐ S'il essaye de l'ouvrir en écriture
- ☐ S'il essaye de l'ouvrir en lecture
- ☐ S'il essaye de communiquer avec des processus n'ayant pas un lien de parenté
- ☒ S'il essaye de l'ouvrir en lecture sans écrivains



☒ S'il essaye de l'ouvrir en écriture sans lecteurs



Votre réponse est correcte.

Les réponses correctes sont :

S'il essaye de l'ouvrir en écriture sans lecteurs,

S'il essaye de l'ouvrir en lecture sans écrivains

Question **31**

Terminer

Non noté

Justifiez votre choix de réponse (Si votre justification est absente ou erronée vous aurez **0 point** pour cette question que votre réponse dans la QCM soit bonne ou mauvaise)

Pour qu'un processus passe en état bloqué, il faut que quand il essaye d'accéder en lecture et qu'il n'y a pas un autre processus en mode écriture ou quand il essaye d'accéder en écriture et qu'il n'y a pas de processus en lecture.

Question **32**

Partiellement correct

Note de 0,80 sur 1,00

Dans le problème du producteur-consommateur :

☐ Deux sémaphores sont utilisés

☒ Le consommateur est mis en attente, s'il veut consommer et le tampon est vide



- ☐ Le producteur est mis en attente, s'il veut produire et le tampon est plein ✓
- ☒ Un processus dépose des objets dans le tampon ✓
- ☒ Un processus retire des objets dans le tampon ✓

Votre réponse est partiellement correcte.

Vous en avez sélectionné correctement 4.

Les réponses correctes sont :

Deux sémaphores sont utilisés,

Un processus retire des objets dans le tampon,

Un processus dépose des objets dans le tampon,

Le producteur est mis en attente, s'il veut produire et le tampon est plein,

Le consommateur est mis en attente, s'il veut consommer et le tampon est vide

Question **33**

Terminer

Non noté

Justifiez votre choix de réponse (Si votre justification est absente ou erronée vous aurez **0 point** pour cette question que votre réponse dans la QCM soit bonne ou mauvaise)

Le producteur va produire des objet dans les cases vides du tampon, en accédant a un 2 des 3 sémaphores, il va produire tant qu'il y a une place libre. Le consommateur va consommer tant qu'il y a un objet dans le tampon. Si le producteur détecte avec le sémaphore que le tampon est plein, il va être bloquer, si le consommateur détecte avec l'autre sémaphore que le tampon est vide il va être bloqué.

## Question 34

Partiellement correct

Note de 0,33 sur 1,00

Considérez quatre processus P1, P2, P3 et P4 ainsi que deux sémaphores dont les valeurs initiales sont S1 = 1 et S2 = 0. Les processus sont lancés en concurrence et effectuent les séquences d'opérations suivantes :

P1	P2	P3	P4
P(S2);	P(S2);	P(S1);	P(S2);
a;	b;	d;	f;
V(S1);	c;	e;	V(S2);
	V(S2);	V(S2);	

Considérant que les opérations a, b, c, d, e et f sont atomiques :

☒ L'ordre d'exécution d, e, b, c, f, a est réalisable



☒ L'ordre d'exécution d, e, f, b, c, a est réalisable



☒ L'ordre d'exécution d, e, b, f, c, a est réalisable



☐ Aucune de ces réponses

☐ Il existe un ordre d'exécution qui provoque un interblocage

Votre réponse est partiellement correcte.

Vous en avez sélectionné correctement 2.

Les réponses correctes sont :

L'ordre d'exécution d, e, f, b, c, a est réalisable,

L'ordre d'exécution d, e, b, c, f, a est réalisable,

Il existe un ordre d'exécution qui provoque un interblocage

## Question 35

Terminer

Non noté

Justifiez votre choix de réponse (Si votre justification est absente ou erronée vous aurez 0 point pour cette question que votre réponse dans la QCM soit bonne ou mauvaise)

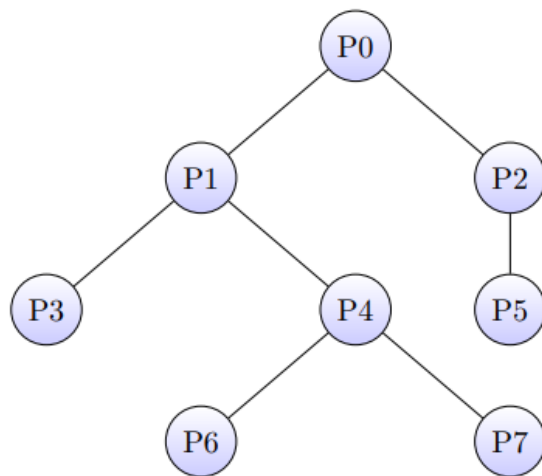
vu que le Sémaphore S1=1 on est obligé de commencer avec lui donc forcément d et e est en avant. après leur exécution P(S2) va incrémenter donc b,c,f vont pouvoir s'exécuter. a doit être exécuter à la fin pour être sûr que tous les opérations sont exécutées.

Question **36**

Terminer

Note de 5,00  
sur 5,00

Écrivez le code nécessaire pour reproduire cet arbre de processus



```
int main() {  
  
    if(fork()==0) {  
        if(fork()==0) {  
            exit(0);  
        }  
        if(fork()==0) {  
            if(fork()==0) {  
                exit(0);  
            }  
            if(fork()==0) {  
                exit(0);  
            }  
        }  
        wait(NULL);  
    }  
}
```



```

        exit(0);
    }
    wait(NULL);
    exit(0);
}

if(fork()==0) {
    if(fork()==0) {
        exit(0);
    }
    wait(NULL);
    exit(0);
}

while(wait(NULL)>0);
exit(0);

```

Commentaire :

### Question 37

Terminer

Note de 3,00  
sur 10,00

Vous devez créer 10 threads niveau noyau. Chaque thread devra exécuter la fonction suivante :

```
void* action(char* nom)
```

- Supposez que vous avez accès aux noms des threads avec un tableau nommé NOMS qui possède 10 entrées.

**Note : sentez-vous libre de créer des fonctions et/ou structures.**

Votre code devra être découpé en trois parties (que vous allez identifier avec un commentaire, exemple //Partie 1 //Partie 2 et //Partie 3)

Il n'est pas obligatoire de donner un code compilable. On vous demande seulement de lister les appels nécessaires, les boucles, conditions, etc. sous la forme de code C.

Partie 1 : (2 pts)

- Vous devez initialiser le sémaphore à 1. Vous n'avez pas besoin de faire l'allocation de la mémoire avec calloc, seulement l'appel d'initialisation. Supposez que vous avez accès à la variable sem\_t\* semaphore que vous utiliserez comme sémaphore binaire.

Partie 2 : (2 pts)

- Vous devez détruire le sémaphore nommé semaphore. Vous n'avez pas besoin de faire l'appel free, on vous demande seulement d'appeler la fonction pour le détruire.

Partie 3 : (6 pts)

- Vous devez créer les 10 threads et faire en sorte que chaque thread appelle la fonction demandée.
- Vous devez définir la fonction action qui, essentiellement, appelle une fonction usageVariablePartagee qui prend comme paramètre le nom. L'appel à la fonction usageVariableProtege est une section critique ! Vous devez donc utiliser le sémaphore pour protéger l'appel à cette fonction.

```
// partie 1
```

```
sem_t* semaphore
pthread_t tids[10];
int main() {
    sem_init(&semaphore,0,1);
    for (i = 0; i < 10; ++i) {
        pthread_create(&tids[i], NULL, action, &NOMS[i]);
    }
    for (i = 0; i < 10; i++) {
        pthread_join(tids[i], NULL);
    }
}
```

// partie 2

```
sem_t* semaphore
pthread_t tids[10];
int main() {
    sem_init(&semaphore,0,1);
    for (i = 0; i < 10; ++i) {
        pthread_create(&tids[i], NULL, action, &NOMS[i]);
    }
    for (i = 0; i < 10; i++) {
        pthread_join(tids[i], NULL);
    }
    sem_destroy(&semaphore);
}
```

// partie 3

Commentaire :

-4 mauvais usage sem\_init et sem\_destroy +1

-6 partie 3 + 1

+1 1er param pthread\_create

### Question 38

Terminer

Note de 8,00  
sur 10,00

Écrivez le code nécessaire pour réaliser les actions suivantes :

- Créer un tube nommé dont le nom (et le chemin) est : /tmp/examen/pipe et l'ouvrir en écriture (supposez qu'un autre process va l'ouvrir en lecture. Ignorez le blocage)
- Rediriger l'entrée 2 de la table de descripteurs de fichiers vers l'entrée 1.
- Rediriger l'entrée 1 de la table de descripteurs de fichiers vers l'entrée 0.
- Rediriger toutes les entrées pointant vers stdout vers le tube
- Transformer le code du processus en exécutant le programme allo prenant en paramètre le PID du processus père.
- Le programme allo se trouve dans le répertoire courant.
- Le répertoire courant n'est pas enregistré dans le PATH.

```
int main() {  
  
    mkfifo("/tmp/examen/pipe", 0666);  
    if(fork()==0) {  
        int f2=open("pipe",O_RDONLY);  
    }  
    else {  
        int f1=open("pipe",O_WRONLY);  
        dup2(1,2);  
        dup2(0,1);  
        dup2(f1[1],2);  
        int pid=getpid();  
        char buffer[30];  
        sprintf(buffer,"%d",getppid());  
        execl("allo", "allo", pid);  
    }  
    return 0;  
}
```

Commentaire :  
mauvais param exec -2

◀ Test Formatif - Chapitre 5 (partie  
1)

Aller à...

Chapitre 5 (partie 2) ▶