ORDRE DES INGÉNIEURS DU QUÉBEC
MAY 2018 SESSION

All documentation allowed
Calculators: Authorized models only
Exam duration: 3 hours

14-IF-A5 Operating systems

QUESTION 1 [20 points]

Concepts of operating systems

(a) [8 points]

i. Transition from the user mode to the kernel mode is a tricky part of an operating system. Explain what happens when an event triggers this transition.
ii. In most modern computers, at least one part of event handling is written in assembly language. Why?

(b) [4 points]

Explain the concept of re-entrancy.

(d) [8 points]

For each of the following instructions, which ones are protected (can be executed only in kernel mode)?

i. Modify the PC (program counter) register
ii. Modify the SP (stack pointer) register
iii. Modify the register that controls kernel/user mode
iv. Direct access to I/O devices

QUESTION 2 [20 points]

Processes, threads and synchronization

(a) [4 points]

Consider the following code:

```
for (i=0 ; fork() ; i++) {
    if (i==6) {break ;}
    printf(''PID : %d, i = %d\n'', getpid(), i) ;
}
```

Suppose that the initial process ID is 1234 and the PID always increments by 1 when a new process is created. Suppose that no new other processes are created during the lifespan of the process.

i. What are the possible outputs of this program?
ii. During the lifespan of the process, how many times is the function fork() called ?

(b) [6 points]

Most modern operating systems provide support for both threads and processes.

i. What is a process?
ii. What is a thread?

(c) [4 points]

A program containing a race condition will always result in data corruption or some other incorrect behavior? Explain.

(d) [6 points]

Explain de following terms:

i. Process control block
ii. User-Level thread
iii. Threads pool

QUESTION 3 [20 points]

Scheduler

(a) [3 points]

In which conditions does FIFO scheduling result in the shortest possible average response time? Explain briefly.

(b) [3 points]

In which conditions does the round-robin scheduler have the same behavior as the FIFO algorithm?

(c) [2 points]

Explain how the approach of aging solves the starvation problem.

(d) [12 points]

Consider the following processes:

Process	Starting time (ms)	Duration(ms)
A	0	20
B	4	10
C	10	25
D	17	10
E	25	5

For each of the following scheduling algorithms, you have to:

1. *Draw the Gantt chart*
2. *Calculate the average waiting time*

i. Shortest job first, no preemption
ii. Shortest job first, with preemption
iii. Round-Robin scheduler with a quantum of 8 ms. Assume that new processes are added at the end of the waiting list.

QUESTION 4 [20 points]

Deadlock

(a) [8 points]

Consider these pieces of code:

Thread 1:
Lock(L1)
Lock(L2)
// critical section
Unlock(L2)
Unlock(L1)

Thread 2:
Lock(L3)
Lock(L1)
// critical section
Unlock(L1)
Unlock(L3)

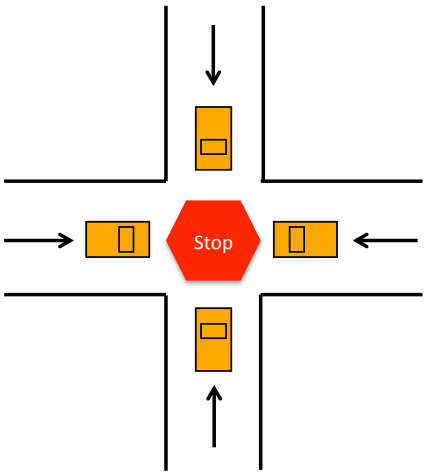
Thread 3:
Lock(L2)
Lock(L3)
// critical section
Unlock(L3)
Unlock(L2)

In the beginning, all locks are initialized as “unlock”. Assume that threads can be executed in any arbitrary interleaving.

Will there be a problem with this code? If yes, show it and explain how the problem can be resolved considering that the use of three locks is mandatory. If there is no problem, explain why.

(b) [12 points]

Consider the following situation where stop signs are present in each direction.



i. Does the rule « yield to the car on right » prevent deadlock? Explain.
ii. Are the 4 conditions to create a deadlock met in this example? Explain.
iii. What is the difference between deadlock and starvation?

QUESTION 5 [20 points]

Memory management

(a) [12 points]

For each of the following pairs of terms, define each of them and clarify the key difference(s) between them.

i. Page and frame
ii. Pagination and segmentation
iii. Reference bit and dirty bit
iv. Internal and external fragmentation

(b)[8 points]

A memory pagination management system uses 4 frames. Given the following sequence of pages:

{ a, b, c, d, e, d, c, b, a, b, c, d, e, f, d, c, b, a, b, c, d, e, f, g, f, e, d, c, b, a }

i. Using the FIFO page replacement algorithm, show the physical memory content after each reference and give the number of page faults.

ii. Using the LRU (*Least Recently Used*) page replacement algorithm, show the physical memory content after each reference and give the number of page faults.