NATIONAL UNIVERSITY OF SINGAPORE

SCHOOL OF COMPUTING
SEMESTER EXAMINATION FOR
Semester 1 AY2010/2011

CS2106 Introduction to Operating Systems

November 2010 Time Allowed 2 hours

MATRICULATION NUMBER:									

INSTRUCTIONS TO CANDIDATES

1. This examination paper contains 15 questions and comprises 12 printed pages, including this page.
2. Answer **ALL** questions.
3. Write **ALL** your answers in the box provided. Please indicate clearly (with an arrow) if you use any space outside the box for your answer.
4. This is a **CLOSED BOOK** examination, but you are allowed to bring in **one sheet of double-sided A4 size paper** with notes.
5. Write your matriculation number in the space provided above and on top-left corner of every page.

EXAMINER'S USE ONLY		
Question	Mark	Score
Q1-10	40	
Q11	12	
Q12	12	
Q13	12	
Q14	12	
Q15	12	
TOTAL	100	

Part I

Multiple Choice Questions (40 points)

For each of the question below, select the most appropriate answer and **write your answer in the answer box**. Each question is worth 4 point.

If none of the answers provided is appropriate, put an X in the answer box. If multiple answers are equally appropriate, pick one and write the chosen answer in the answer box. **Do NOT write more than one answers in the answer box.**

1. System calls are expensive. Different strategies have been used to reduce the overhead of making system calls. Which of the following is the LEAST appropriate strategy for reducing the overhead of a system call?

   A. keep the kernel as small as possible and move certain OS components into user space.

   B. wrap system calls with library procedure that does the necessary caching so that subsequent calls will simply return the cached value.

   C. run shared libraries commonly used by user applications in kernel mode.

   D. batch multiple system calls into one call.

   Answer: □

2. Consider the following code fragment:

```
pid_t pid = fork();
printf("%d\n", pid);
if (pid == 0) {
        execlp("ls", "ls");
        printf("%d\n", pid);
}
```

   Which of the following will NOT be printed by the functions `printf` above?

   A. 0

   B. process ID of the parent process

   C. process ID of the child process

   D. process ID of the process `ls`

   Answer: □

3. Context switch happens when the following events occur, EXCEPT:

   A. when a process calls `exit`

   B. when a process calls `read`

   C. when a process calls `brk`

   D. when a process encounters a major page fault

Answer: □

4. Compare the following two scenarios. In Scenario A, there is only one process P running in the system. In Scenario B, there are two processes, P and Q, which take turn to run.

   P may take a longer time to run to completion in Scenario B compared to Scenario A due to the following reasons, EXCEPT:

   A. P may encounter more page faults

   B. P may have to wait for Q to enter a critical region

   C. P may have to wait for a resource held by Q

   D. P may take longer to read data from disk.

Answer: □

5. Consider the following snapshot of a system's states, with three processes and four allocatable resources. The current allocated and requested resources are as follows:

		Available
		0 0 1 1
	Allocated	Requested
Process A	2 1 1 1	1 0 0 0
Process B	1 1 0 1	0 2 x 2
Process C	1 1 y 1	0 0 1 0

   x and y are two unknowns. What should the relationship between x and y be to ensure all three processes can run to completion?

   A. $x \leq y - 1$

   B. $x \geq y - 2$

   C. $x \leq y + 2$

   D. $x = y$

Answer: □

6. A computer system has a physical memory that consists of four frames. Assuming that the physical memory is initially empty, how many page fault would the following page access sequence generates if LRU page replacement algorithm is used? (Remember to include the page fault generated the first time a page is accessed.)

   1, 2, 3, 4, 5, 1, 2, 3, 4, 5

   A. 1

   B. 5

   C. 6

   D. 10

   Answer: □

7. An embedded system uses 16-bit for memory addresses. Each memory page is 512 bytes. How many page entries are there in the page table, if a non-hierarchical page table is used?

   A. 2^{25}

   B. 2^{16}

   C. 2^{9}

   D. 2^{7}

   Answer: □

8. Which of the following items are NOT stored on the memory stack (stack segment) when making a function call to function f?

   A. frame pointer of the function that calls f

   B. program counter of the instruction to execute after returning from f

   C. local variables of f

   D. arguments to the function f

   Answer: □

9. Which of the following statement is FALSE?

   A. Page table maps a page in virtual memory to a frame in physical memory.

   B. Process table keeps track of all processes in the system.

   C. An i-node maps a file name to first disk block of that file.

   D. File description table keep tracks of open files of a process.

   Answer: □

10. A page fault (major or minor) is LEAST likely to be triggered when the following events occur:

    A. a memory location that has just been allocated with `malloc` via anonymous memory map is written to for the first time.

    B. the CPU performs a context switch to a newly created process.

    C. a process invokes a system call for the first time.

    D. a shared library is used for the first time by any process.

Answer: □

Part II

Short Questions (60 points)

Answer all questions in the space provided. Be succinct and write neatly.

11. (12 points) A start-up company came up with the idea of providing email services with unlimited storage so users never need to delete their emails (in fact, users cannot delete their emails). The company has decided that every email sent to a user would be stored in one huge file, in the order that the emails are received. Thus, the file only grows in size (files in order of GBs are normal). New emails sent to a user are appended to the file corresponding to that user. Further, users are most likely to access new and recent emails, and only rarely read older emails.

    The company would like to design a customized file system to store files containing emails from users. Suggest two changes to a conventional file system (either i-node or FAT) that would make access to emails more efficient.

    ....

    ....

    ....

    ....

    ....

    ....

    ....

    ....

    ....

    ....

    ....

    ....

    ....

    ....

    ....

    ....

    ....

    ....

    ....

    ....

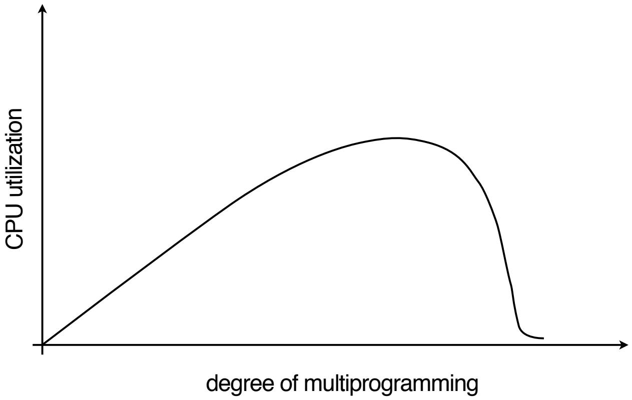
    ....

    ....

12. (12 points) The following figures plot how CPU utilization changes with the degree of multiprogramming.

    For each of the following changes to the system (made independently of each other), *sketch a new curve on the same figure* to illustrate how CPU utilization would change. Explain why the curves have shifted in the way you have illustrated.

    (a) Increase the size of physical memory



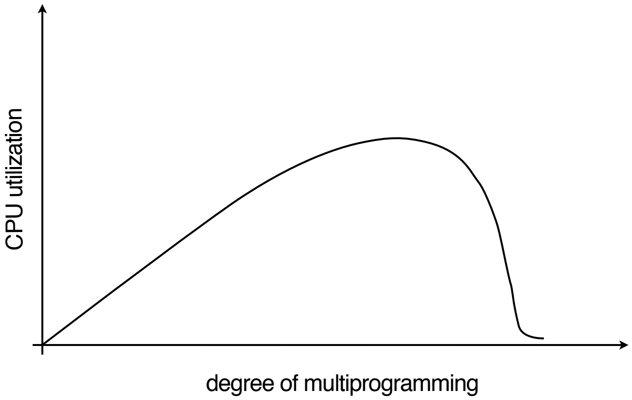
....

....

....

....

    (b) Upgrade to faster CPU



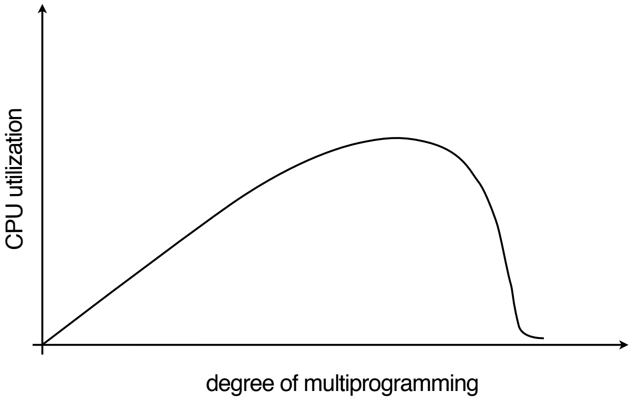
....

....

....

....

(c) Increase the size of swap space



. . . .

. . . .

. . . .

. . . .

13. (12 points) A buggy piece of code tried to enter the critical region from within the critical region (i.e., the code calls `enter()` twice without calling `leave()`). For each of the following implementation of `enter()` and `leave()`, explain what would happen when the code calls `enter()` the second time.

    (a) disabling and enabling interrupts.

    ....

    ....

    ....

    ....

    ....

    ....

    (b) Peterson's algorithm.

    ....

    ....

    ....

    ....

    ....

    ....

    (c) atomic test-&-set lock.

    ....

    ....

    ....

    ....

    ....

    ....

14. (12 points) Consider a generalization of critical region, called *class-based critical region*, described as follows. There are two classes of processes, numbered 0 and 1. Processes of either class can enter critical region. But if a Class-0 process is running within a critical region, only other Class-0 processes can enter. Class-1 processes must wait, until all Class-0 processes have exited the critical region. Similary, if a Class-1 process is running within a critical region, only other Class-1 processes can be inside the critical region.

    In other words, such critical region allows multiple processes to be inside the region, but only if they are of the same class.

    Processes call the functions `enter_class(class)` and `leave_class(class)` respectively before they enter and leave a critical region, where the parameter `class` is the class of the calling process (either integer 0 or 1).

    Sketch the pseudocode of `enter_class(class)` and `leave_class(class)` implemented using semaphores. Indicate the initial values of your semaphores clearly. Note that only two operations are allowed on semaphores: up() and down().

15. (12 points) Consider an operating system that uses multi-level feedback queue (MLFQ) for scheduling. To improve the average response time of interactive jobs, some modifications to the scheduling algorithm have been suggested.

    The following lists the suggested modifications (to be made independently of each other) to the scheduling algorithm. For each suggested modification, would average response time of interactive jobs be improved? Explain your answers clearly, stating any assumption you make about the working of the MLFQ algorithm.

    (a) Reduce time quantum of each job by half.

    (b) When a job blocks for I/O, instead of keeping the job at the same queue, move the job to the highest priority queue.

(c) Instead of assigning larger time quantum to jobs with lower priorities, use the time quantum of the highest priority job as the time quantum of all jobs (regardless of priority).

....

....

....

....

....

....

....

....

(d) Instead of using round-robin to schedule jobs in each queue, use shortest remaining time first to schedule jobs in each queue.

....

....

....

....

....

....

....

....

END OF PAPER