Matriculation Number: □□□□□□□□□ CS2106

NATIONAL UNIVERSITY OF SINGAPORE

SCHOOL OF COMPUTING
SEMESTER EXAMINATION FOR
Semester 1 AY2011/2012

CS2106 Introduction to Operating Systems

NOVEMBER 2011 Time Allowed 2 hours

INSTRUCTIONS TO CANDIDATES

1. This exam paper contains 14 questions and comprises 10 printed pages, including this page.
2. The total marks for this examination is 100. Answer **ALL** questions.
3. Write **ALL** your answers in the lined area provided. Please indicate clearly (with an arrow) if you use any space outside the lined area for your answer.
4. This is an **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 on the top-left corner of every page.

EXAMINER'S USE ONLY		
Q1-10	30	
Q11	18	
Q12	20	
Q13	15	
Q14	17	
TOTAL	100	

Part I

Multiple Choice Questions (30 points)

For each of the question below, select the most appropriate answer and **write your answer in the answer box**. Each question is worth 3 points.

It is possible that none of the answers provided are appropriate. If you believe that NONE of the answers are appropriate, put an `X` in the answer box.

It is also possible that multiple answers are equally appropriate. In this case, pick ONE and write the chosen answer in the answer box. Do NOT write more than one answers in the answer box.

1. In an OS that is based on monolithic kernel architecture, which of the following handler does NOT always execute in kernel mode?

   A. Signal handler

   B. System call handler

   C. Interrupt handler

   D. Page fault handler

   Answer: □

2. Which of the following is NOT a resource allocation strategy that prevents deadlock?

   A. Ensure that a resource is only assigned to one process at a time.

   B. Ensure that a process acquires resources only when every resource needed is available.

   C. Allow a process to preempt a resource held by another process.

   D. Allow a process to release a resource it acquired before using it.

   Answer: □

3. There are three processes A, B, and C, and three non-preemtable resource types (one copy of each type) R, S, and T in the system.

   A needs to acquire R and S; B needs to acquire all three R, S, and T; and C needs to acquire R and T.

   Which of the following statement is FALSE?

   A. If A has acquired S, B has acquired R, and C has acquired T, then all three processes A, B, and C are in deadlock.

   B. If A has acquired R, B has acquired S and T, then all three processes A, B, and C are in deadlock.

   C. If A has acquired S, B has acquired T, and C has acquired R, then all three processes A, B, and C are in deadlock.

   D. If R is always acquired before S and S is always acquired before T, there will never be a deadlock.

   Answer: □

4. Consider a system that only admits two types of tasks, short tasks (each one takes 1 unit time to complete) and long tasks (each one takes 10 unit time to complete). The tasks arrive in random order and are completely CPU-bound.

   The system uses Round-Robin (RR) scheduler to schedule the tasks.

   Which of the following statement is TRUE?

   A. compare to using First-Come-First-Serve, the short tasks tend to complete earlier in RR

   B. compare to using First-Come-First-Serve, the short tasks have a longer response time in RR

   C. compare to using Shortest-Job-First, the long tasks tend to complete later in RR

   D. compare to using Shortest-Job-First, the long tasks have a longer response time in RR

   Answer: □

5. Two processes can communicate with each other using the following system calls, EXCEPT:

   A. mmap()

   B. pipe()

   C. kill()

   D. exec()

   Answer: □

6. You are trying to figure out what page replacement algorithm a particular system is using. Suppose the system has 4 memory frames, initially empty. The sequence of read-only pages referenced are: 1, 2, 3, 4, 1, 5. A clock tick resets all R bits to 0 after Page 3 is referenced. When Page 5 is referenced, Page 2 is evicted from the memory.

   Which of the following algorithm CANNOT possibly be the page replacement algorithm used?

   A. LRU

   B. FIFO

   C. Clock

   D. NRU

   Answer: □

The next two questions is based on the system description below.

A computer system has 4 GB of physical byte-addressable memory with 32 bit memory addresses. It uses virtual memory with demand paging for memory management, with a page size of 4 KB. Each page entry contains a page ID, a frame ID, and three additional bytes of information.

7. How many bits of the virtual address are replaced by the MMU when it translates the address into a physical address?

   A. 12

   B. 16

   C. 20

   D. 32

   Answer: □

8. How big is the page table, assuming that a single level page table is used?

   A. 128 KB

   B. 1 MB

   C. 8 MB

   D. 4 GB

   Answer: □

The next two questions are based on the systems description below.

Consider a UNIX-like file system implemented with i-nodes that resides on a disk of size 512 GB. Each i-node has a total of 15 block addresses, consisting of direct and indirect block addresses.

9. Suppose we configure the file system to use a block size of 32 KB. How many bytes are needed to store all 15 block addresses in an i-node?

   A. 15 bytes

   B. 29 bytes

   C. 45 bytes

   D. 74 bytes

   Answer: □

10. Suppose the implementation wants to support file sizes up to 1 GB using only direct block addresses and single indirect block addresses.

    At least how many of the 15 block addresses should be used as single indirect block addresses?

    A. 0

    B. 1

    C. 2

    D. 3

    Answer: □

Part II

Short Questions (70 points)

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

11. (18 points) For each type of table structure below, indicate whether the OS maintains one such table per process or one table for the whole system. If there is one table per process, write "P" in the corresponding column; otherwise, write "1" in the column.

Further, for each table structure, indicate what the table is *mainly* used for (to map from what to what).

The answer for the first entry has been given to you as a sample.

Table	P or 1	Map From	Map To
Interrupt Vector Table	1	interrupt number	interrupt handler
Page Table			
Process Table			
File Descriptor Table			
i-node Table			
Signal Handler Table			
File Allocation Table			

12. (20 points) **The Dining Couple Problem**. A couple, Romeo and Juliet, share a bowl of frozen yogurt[1]. Only one of them can eat from the bowl at a time, and they must take turn to eat. It does not matter who eat first.

    Each of the following incorrect solutions uses semaphores to model the behavior of the couple.

    (a) (5 points)

```
semaphore s  = 1;
semaphore romeo = 0;
semaphore juliet = 0;

Romeo:                 Juliet:
  while (true) {         while (true) {
    down(s)                down(s)
    eat()                  eat()
    up(juliet)             up(romeo)
    down(romeo)            down(juliet)
    up(s)                  up(s)
  }                      }
```

Give a sequence of execution whereby the pseudo-code above would lead to deadlock.

....

....

....

....

....

....

....

....

....

....

[1] We assume that the bowl is magical and has an infinite supply of frozen yogurt.

(b) (5 points)

```
semaphore s  = 1;
semaphore romeo = 0;
semaphore juliet = 0;

Romeo:                  Juliet:
  while (true)  {         while (true) {
    down(s)                 down(s)
    eat()                   eat()
    up(s)                   up(s)
    up(juliet)              up(romeo)
    down(romeo)             down(juliet)
    }                     }
```

Give a sequence of execution whereby the pseudo-code above would lead to Romeo eating twice consecutively without passing the bowl to Juliet.

....

....

....

....

....

....

....

....

....

....

(c) (10 points) Write the correct solution (in pseudo-code) that models the behavior of Romeo and Juliet as described above using only semaphores. In your solution, let Romeo be a gentleman – so Juliet gets to have the first bite.

You only need to permute the order of the statements in the pseudo-code given in Part (a) or (b) and change the initial value of one of the semaphores.

13. (15 points) Consider a system with the following behavior:

- accessing (either read or write) an entry in the TLB takes T seconds;
- accessing (either read or write) an entry in the page table takes P seconds;
- fetching a value from the physical memory takes M seconds;
- disk I/O (either read or write) takes D seconds per page;

Assume that a single-level page table is used. Ignore all other costs (such as context switching) in your answers below and consider only the costs T, M, P, and D.

(a) (5 points) In the BEST case, how long does it take to fetch a value from the physical memory given its virtual memory location? Express your answer in terms of the costs above.

Explain each term in your answer.

(b) (10 points) In the WORST case, how long does it take to fetch a value from the physical memory from a virtual memory location? Express your answer in terms of the costs above.

Explain each term in your answer.

14. (17 points) The following simple C program allocates n bytes of memory and writes to the content of the allocated memory repeatedly for 10,000 times.

```
#include <stdlib.h>
#include <string.h>
int main(int argc, char *argv[])
{
  long n = atoi(argv[1]); // n is a command line argument
  char *x = malloc(n);    // allocate n bytes of memory
  for (int i = 1; i <= 10000; i++)
    for (int j = 0; j < n; j++)
      x[j] = 1;           // set the allocated memory to ones.
}
```

Suppose that we run the process above in a system that uses LRU as the page replacement algorithm (implemented using a queue). Assume that the cost for handling page fault is high. Let F be the number of page faults that occur due to the assignment operation `x[j] = 1`.

We now run the process above with different memory sizes n, ranging from smaller than the size of a page to larger than the total amount of physical memory available. Let p be the size of a page (in bytes), and M be the total number of bytes that can be allocated to the process above in physical memory (i.e., excluding the region occupied and used by the kernel). Assume that the process has exclusive right to run on the machine and no other processes can interrupt the process until it terminates.

(a) (5 points) What is F if n is less than the size of a page p? Justify your answer.

....

....

....

....

(b) (5 points) What is F if n is less than M? Express your answer in terms of n and p. Justify your answer.

....

....

....

....

(c) (7 points) What is F if n is larger than M? Express your answer in terms of n and p. Justify your answer.

....

....

....

....

END OF PAPER