Operating Systems
Final Exam, Fall 2013

Date: Dec 24, 2013 **Marks:** 100 **Time:** 3 hours

Question 1 (20 marks)

Consider robots moving up and down a ladder. Assume the ladder is narrow and hence only one robot can climb up or down at a time. However, multiple robots can move in one direction at the same time. All the robots are connected to a system where they can access data common to the ladder and can create common mutexes and semaphores.

Give pseudocode for such a moving robot. Synchronize the climbing using Semaphores and Mutexex. Do not worry about starvation: assume the train of robots is not very long.

Question 2 (20 marks)

Consider the following page table of a process P: (All numbers are in hexadecimal)

Page	0	1	2	3	4	5	6	7
Frame	9F	5C	1A	4D	7E	11	FF	66

The size of the address register is 16 bits and the page size is $(100)_{16}$. Convert the following logical addresses generated by P into the physical addresses:

i. 008E
ii. 061A
iii. 0357
iv. 07BC
v. 02D7

Question 3 (10 + 5 = 15 marks)

a) Show execution of the Optimal Page replacement algorithm on the following page reference string. Assume only three frames are available. If the Optimal algorithm is unable to guide at some point then use FIFO.

1 2 3 4 1 5 2 6 3 4

b) Write down the steps of handling a page fault in a virtual memory system.

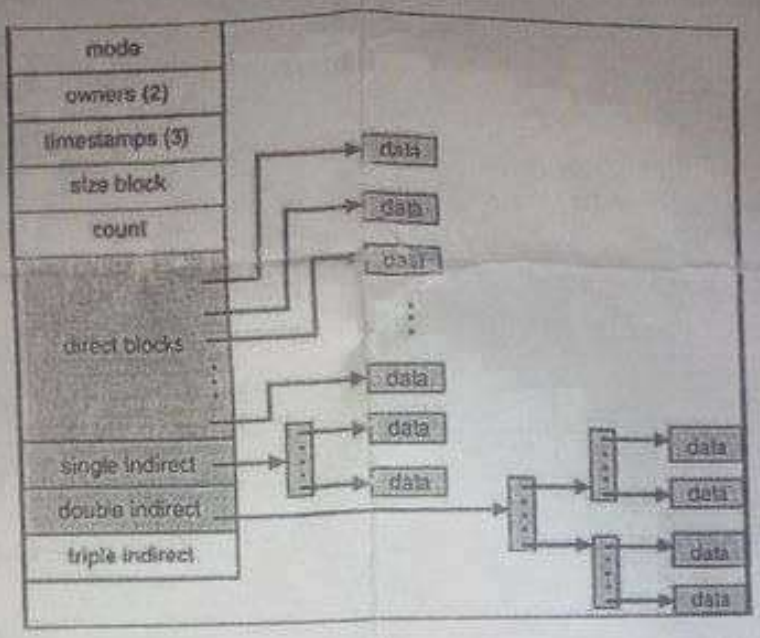
Queston4 (10 + 5 = 15 marks)

a) In a text editor we need two computational units: one to perform editing and another to perform spell-checking. What would you use: (1) two processes or (2) two threads? Give at least two reasons for your choice.

b) A process running a file "a.exe" wants its child to run a file "b.exe". What would the child be a new process or a new thread? Give reason for your answer.

Question 5 (15 marks)

Consider an operating system using Indexed Allocationfor storage of files:



Give a C/C++ function to compute the physical block number of a file from a given logical block number in such a system. Assume the INODE is read into the following C structure and there is a function `read(long block)` which can read any data block on a volume. Also assume that the block size is 4k bytes.

```
structINode {
        // ...
        long block[12];  // addresses (numbers) of the first 12 data blocks
        long indirect;   // address of the single indirect block
        longdblInd;      // address of the double indirect block
        longtplInd;      // address of the triple indirect block
};
```

Following is the prototype of your function. It takes a file's inode and a logical block number, and returns the physical block number: `long map(INodeinode, longlogBlk)`

Question 6 (15 marks)

Consider an operating system that recalculates the process priorities once per second using the following formula:

priority = (recent CPU usage / 2) + 60 (the higher the number, the higher the priority)

a) What would happen to an I/O bound process, and what would happen to a CPU bound process?

b) Is this treatment good or bad in an interactive environment? Why or why not?

c) Can you improve the above algorithm? If yes how would you do that?