國立台灣科技大學九十七學年度碩士班招生試題

組別： 資訊工程系碩士班

科目： 作業系統

總分 100 分

1. Consider a processor executing a move instruction, with both two of its operands possibly use direct or one-level indirect addressing. Assume the computer system is equipped with TLB and uses inverted page table for memory management.
   (a) (4%) Ignoring the page table lookup part, what are the minimum and maximum numbers of pages that have to be referenced in order to complete this instruction?
   (b) (6%) If there is another hash table used to speed up the search operation of the inverted page table, what are the minimum and maximum numbers of memory accesses needed to locate the address of an instruction? Justify your answer.

2. Below is a list of events/steps that happen during a page replacement progress, but these events/steps are not given in order of occurrence:
   A. A TLB miss occurs
   B. A page-fault trap occurs
   C. Search the page table and find the demanding page has its valid bit unasserted
   D. Try to find a free frame from the frame pool, but no free frame remains
   E. Check the frame table for the newly allocated frame. If the dirty bit is set, schedule a disk operation to write the frame back to secondary memory.
   F. Save the user registers and process state, then allocate the CPU to other process
   G. Find a page to be replaced
   H. Restart the postponed process
   I. Schedules a disk operation to read the desired page into the newly allocated frame
   J. When the disk read is complete, updates the page table and TLB
   (a) (5%) Given A as the first event, what is the correct order of other list items?
   (b) (5%) Must a TLB miss occurs before a page fault may take place? Why or why not?
   (c) (2%) If second-chance algorithm is used for page replacement, in which step of the above list should the second-change algorithm be specified?
   (d) (8%) Describe how second-chance algorithm works.

3. (15%) Suppose that a disk drive has 4860 cylinders, numbered 0 to 4859. The

國立台灣科技大學九十七學年度碩士班招生試題

系所組別： 資訊工程系碩士班

科　　目： 作業系統

drive is currently serving a request at cylinder 143, and the previous request was at cylinder 125. The queue of pending requests, in FIFO order, is 86, 1470, 913, 1774, 948, 130. Starting from the current head position, what is the total distance (in cylinders) that the disk arm moves to satisfy all the pending requests, for each of the following disk-scheduling schemes?

(a) FCFS
(b) SSTF
(c) SCAN
(d) LOOK
(e) C-SCAN

4. (5%) How does the NTFS directory structure differ from the directory structure used in Unix operating systems?

國立台灣科技大學九十七學年度碩士班招生試題

所組別： 資訊工程系碩士班

目： 作業系統

5. (10%) Consider the following snapshot of a system:

Process ID \ Resource	Allocation				Max				Required			
	A	B	C	D	A	B	C	D	A	B	C	D
1	0	0	1	2	0	0	1	2	0	0	0	0
2	0	1	0	0	7	1	5	0	7	0	5	0
3	3	1	5	4	3	2	5	6	0	1	0	2
4	6	0	3	2	6	0	5	2	0	0	2	0
5	0	0	1	4	6	0	5	6	6	0	4	2

Currently available resource [A, B, C, D] = [5, 1, 2, 0]. Will the system be in a safe state if Job 5's request of [3, 0, 1, 0] is granted at this moment? **Show all steps of the computation or receive no credit.**

6. (5%) What do "`ps`" and "`top`" mean under a Unix/Linux shell?

7. (10%) Can a process transit directly from the waiting state (for I/O operations) to the terminated state? Draw the state transition diagram of process execution to justify your answer.

8. (10%) Is it true that the Multi-Level-Feedback-Queue (MLFQ) scheduling is an approximation to the Shortest-Job-First (SJF) scheduling? Why or why not?

9. (15%) Suppose *N* processes need to share one critical region of code, which only one process is allowed to execute at any time. And the *N* processes are required to indefinitely cycle through this critical region in the order of their process IDs (0...*N*-1). Let `turn` be a shared integer variable initialized to zero and `cpid` be the ID of the currently executing process. Could a race condition occur if each process executes the following code? Why or why not?

```
while( TRUE ) {
  while( turn != cpid ) /* wait */;
  critical_region();
  turn = (turn + 1) % N;
  noncritical_region();
}
```