00055

THE OPEN UNIVERSITY OF SRI LANKA
FACULTY OF ENGINEERING TECHNOLOGY
BACHELOR OF TECHNOLOGY HONOURS IN ENGINEERING
DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING
ECX 5235 – OPERATING SYSTEMS

FINAL EXAMINATION – 2015 / 2016 **(CLOSE BOOK TYPE)**

Date: 29th, November 2016 **Time: 13:30 – 16:30hrs**

INSTRUCTIONS TO CANDIDATE:

i.) This paper contains eight(8) questions. Answer any five(5) questions. All questions carry equal marks.
ii.) Assume reasonable values or any suitable assumptions for any data not given in or if any doubt as to the interpretation of the wording of a question. Clearly state such assumptions made on the script.
iii.) You are allowed to use scientific calculators during the exam.
iv.) You are NOT allowed to use any study material or any other electronic resource during the examination.

Question 1

i.) Briefly explain the impact of the evolution of computer hardware with respect to the evolution of operating systems software.

[04 Marks]

ii.) Briefly explain a situation that might demand a real-time operating system and why.

[04 Marks]

iii.) State how a policy of a *timesharing processor* scheduler might differ from a policy used in a *batch system.*

[08 Marks]

iv.) A program executing in user mode can request the kernel's services using *system call* and *message passing* techniques. Describe the difference between the *system call* and *message passing* techniques.

[04 Marks]

Question 2

i.) Compare *polling-based I/O* with *interrupt-driven I/O.* In what situation would you favor one technique over the other?

[06 Marks]

ii.) What file allocation strategy is most appropriate for random access files? Justify your answer.

[06 Marks]

iii.) Give a scenario where choosing a large filesystem block size might be a benefit and give an example where it might be an interference.

[08 Marks]

Question 3

i.) Assume you have the following jobs to be executed with one processor with the jobs arriving in the order listed here. i^{th} process is represented by p_i and the service time for each process is represented by $T(p_i)$

i	$T(p_i)$	*Arrival Time(milliseconds)*
0	90	0
1	30	10
2	20	20
3	30	80
4	60	85

a) Suppose a system uses First Come First Served scheduling. Draw a Gantt chart illustrating the execution of these processes.

[2 Marks]

b) Calculate the turnaround time for process p_3.

[2 Marks]

c) Calculate the average wait time for the processes.

[2 Marks]

ii.) Suppose a system uses Round Robin(RR) scheduling for the given processes with zero context switching time and a time quantum of 15 milliseconds.

a) Draw a Gantt chart illustrating the execution of these processes.

[2 Marks]

b) Calculate the turnaround time for process p_3.

[2 Marks]

c) Calculate the average wait time for the processes.

[2 Marks]

d) If the context switching time is changed to 0.4 milliseconds, show how the change would affect the turnaround time and average wait time.

[4 Marks]

iii.) Describe the effect of increasing the time quantum to an arbitrary large number for RR scheduling.

[4 Marks]

Question 4

i.) Briefly explain the steps involved in handling a page fault.

[4 Marks]

ii.) Memory paging is a feature that permits extending the address space far beyond the available memory. [X] and [Y] denote different addresses. Name the components given as [X], [Y] and [Z] in the following diagram *(Figure 1)* and describe how the extension of the address space happens.

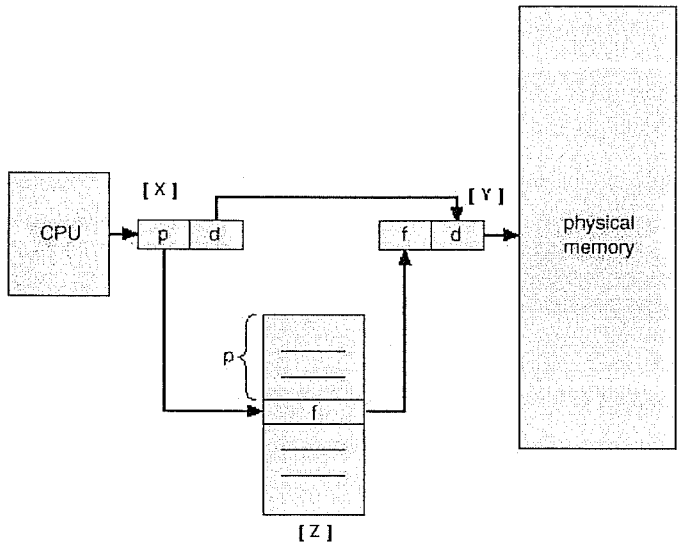

Figure 1: Memory Paging

[6 Marks]

iii.) Page size is one of the parameters of a virtual memory. Briefly describe one advantage and one disadvantage of choosing a large page size rather than a small one.

[4 Marks]

iv.) The following diagram *(Figure 2)* depicts CPU utilization vs degree of multiprogramming. Focus on the regions shown by (A) and (B) and describe the reasons for such a behavior.

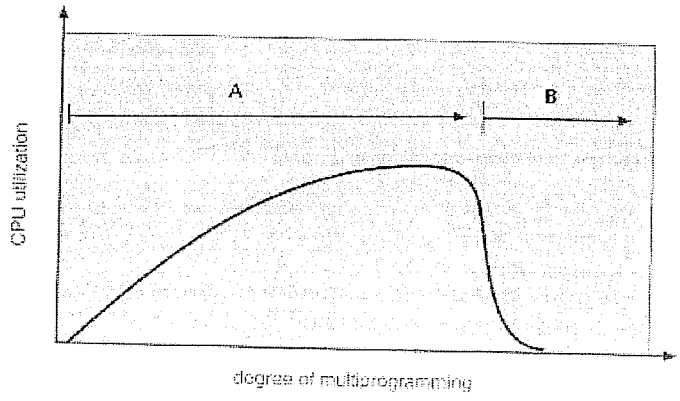

Figure 2: CPU Utilization vs Degree of multiprogramming

[6 Marks]

Question 5

i.) Compare user level thread with kernel level thread.

[4 Marks]

ii.) ***A*** and ***B*** represent two *3x3* matrices. Compare the number of steps it would take to perform matrix multiplication ***C= A * B*** when execute serially with only one processor and parallel with three processors available.

[8 Marks]

iii.) Consider the following program segments for two different processes (P1, P2) executing concurrently and where ***a*** and ***b*** are not shared variables, but ***x*** starts at zero and is a shared variable.

P1	P2
for(a=1;a<=3;a++) x=x+1;	for(b=1;b<=3;b++) x=x+1;

If the processes P1 and P2 execute only once at any speed, what are the possible resulting values of ***x***? Explain your answer.

[8 Marks]

Question 6

i.) Consider the following state of a system at a particular point in time(snapshot) with five processes (P1, P2, P3, P4, P5) and four resources (R1, R2, R3, R4).

Available No. of Resources

R1	R2	R3	R4
2	1	2	0

	Current Allocation				Requests Made			
Process	R1	R2	R3	R4	R1	R2	R3	R4
P1	0	0	1	2	0	0	2	0
P2	2	0	0	0	0	1	1	0
P3	0	0	3	2	2	1	2	2
P4	2	3	2	2	2	0	0	2
P5	0	3	3	2	0	3	2	0

Is this system currently deadlocked, or can any process become deadlocked? Why or why not? Explain using a resource allocation graph.

[06 Marks]

ii.) Briefly explain the four conditions that must be fulfilled for a deadlock to occur.

[04 Marks]

iii.) Assuming the operating system detects the system is deadlocked, state three measures that can be taken by the operating system to recover from deadlock.

[06 Marks]

iv.) State what the Banker's algorithm must know a priori in order to prevent deadlock.

[04 Marks]

Question 7

i.) Briefly explain the use of *Process Control Block* in the operating system kernel.

[02 Marks]

ii.) Draw process state transition diagram and explain each transition of states.

[08 Marks]

iii.) Write a C program which uses Unix/Linux system calls to create a new process. The child process should create an empty file called 'abc' and then terminate. The parent process should wait for the child process to terminate and then output the process number of the child process. You need to check for error conditions. A useful example of C code snippet is given below.

[10 Marks]

The C library function **FILE *fopen(const char *filename, const char *mode)** opens the **filename** pointed to, by filename using the given **mode**.

The following example shows the usage of fopen() function.

```
#include <stdio.h>
#include <stdlib.h>
int main()
{
  FILE * fp;
  fp = fopen ("file.txt", "w+");
  fprintf(fp, "%s %s %s %d", "We", "are", "in", 2012);
  fclose(fp);
  return(0);
}
```

Question 8

i.) Briefly explain the importance of taking necessary measures to ensure security in an Operating System.

[04 Marks]

ii.) Distinguish *Program Threats* from *System threats* using examples for each.

[06 Marks]

iii.) Briefly explain two techniques used in Operating Systems to ensure safety from threats and attacks.

[04 Marks]

iv.) Windows XP is a general purpose OS designed to support a wide variety of security features and methods. It is based on user accounts which can be grouped in any manner. When a user logs on, a security access token is issued. State three advantages of issuing security access tokens.

[06 Marks]