244

B.Sc/HD CIT 5th Semester 8 May, 2011 (Morning)

ISLAMIC UNIVERSITY OF TECHNOLOGY (IUT)
THE ORGANIZATION OF THE ISLAMIC CONFERENCE (OIC)
Department of Computer Science and Information Technology (CIT)

SEMESTER FINAL EXAMINATION **WINTER SEMESTER, 2010-2011**

Duration: 3 Hours **Full Marks: 150**

CIT 4501: Operating Systems Fundamentals

Programmable calculators are not allowed. Do not write anything on the question paper.

There are 8 **(eight)** questions. Answer any 6 **(six)** of them.

Figures in the right margin indicate marks.

1. a) Write short note on Time Shared Operating System. State the difference between Soft and Hard Real time systems. 2+3

   b) What are the differences between process and thread? What are the benefits of Multithreaded process. 4+4

   c) Suppose that a disk drive has 100 cylinders, numbered from 0 to 99. The drive is currently serving a request at cylinder 99, and the previous request was at cylinder 0. The queue of pending requests, in FIFO order is: 12

   86, 47, 91, 17, 98, 15, 12, 17, 23

   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 algorithms?

   i. LOOK
   ii. SSTF
   iii. C-SCAN
   iv. C-LOOK

2. a) What is the purpose of File System Mounting? Describe File System Mounting. 3+5

   b) Write the description as well as the pros and cons of the followings. 5

   i. Two level directory structure
   ii. General graph directory structure

   c) Consider memory access time is 100 nanoseconds, average page-fault service time is 4 milliseconds. If one access out of 500 causes a page fault, then what is the Effective Access Time? 4

   d) Draw the Gantt chart and find the average ***Turn Around Time*** for SJF (non preemptive) and RR Scheduling algorithms. 4+4

Process	Burst Time	Arrival Time
P1	6	0
P2	8	3
P3	12	4
P4	9	7

   Consider Time Quantum = 3 unit.

3. a) What are the process states? Draw the Queuing-Diagram that represents the *Process Scheduling* technique. 2+4

b) Write short notes on the followings: 2+2+3

i. Context switching

ii. CPU scheduler

iii. Medium scheduler

c) Given memory partitions of 200 KB, 600 KB, 200 KB, 400 KB and 700 KB (in order), how would each of the first fit, best fit and worst fit algorithms place for the processes of 215 KB, 420 KB, 115 KB and 430 KB (in order)? Which algorithm makes the most efficient use of memory? 10+2

4. a) State the differences between preemptive and Non preemptive scheduling criteria with example. 2+3

b) Although the Shortest Job First (SJF) algorithm is the best of all CPU scheduling algorithms yet it is not possible to implement practically. Explain the approximated implementable SJF algorithm with example. 6

c) Draw the Gantt chart and find the average waiting time for SJF (pre-emptive), RR, FCFS, Non Pre-emptive Priority Scheduling for the following chart: 3x4+2

Process	Burst Time	Arrival Time	Priority
P1	5	0	2
P2	6	3	4
P3	11	4	3
P4	8	7	0

Consider highest priority number has the lowest priority and Time Quantum = 4 unit. In which case RR is better than the other algorithms?

5. a) Explain the structure of shared pages. 6+2

b) What is paging? Describe paging hardware with diagram. 3+2

c) Consider a paging system with TLB and hit ratio of 80%. If the memory access time is 150 nanoseconds and the searching time for TLB is 50 nanoseconds then what is the *effective memory access time* for the following cases? 2+2.5

i. When the page-table entry is found in the TLB

ii. When the page-table entry is not found in the TLB

d) Consider the following segment table: 7.5

Segment	Base	Length
0	219	600
1	2300	14
2	90	100
3	1327	580
4	1952	96

What are the physical addresses for the following logical addresses?

i. 4, 112

ii. 2, 500

iii. 3, 400

iv. 0, 430

v. 1, 10

6\. a) What is semaphore and why is it used? How is it possible to evaluate mutual exclusion using binary semaphore? Write the structure. 3+4

b) Describe the Multithreaded models. 6

c) Write the structures of Reader Writer processes with the appropriate semaphore definitions. 6

d) What will be the output of the following program? You must maintain the execution order of parent and child processes. 3+3

```
#include <iostream>
#include <string>
#include <sys/types.h>
#include <unistd.h>
#include<stdlib.h>

using namespace std;
int globalVariable = 2;

main()
{
  string sIdentifier;
  int    iStackVariable = 80;

  pid_t pID = fork();
  if (pID == 0)
  {
    sIdentifier = "Child Process: ";
    for(int i=0;i<100;i++){
      globalVariable++;
    }
    for(i=10;i>0;i--){
       iStackVariable--;
    }
  }
  else if (pID < 0)
  {
     cerr << "Failed to fork" << endl;
     exit(1);
  }
  else
  {
    sIdentifier = "Parent Process:";
  }

  cout << sIdentifier;

  cout << "Stack variable:"  << iStackVariable << endl;
  cout << "Global variable:" << globalVariable <<endl;
  return 0;
}
```

247

7. a) Explain how it is possible to calculate Wait For Graph. Why is this Wait For Graph necessary? 4

b) Write the algorithm of deadlock detection of several instances of resource type. 2+3

c) Differentiate ***Starvation*** and ***Deadlock*** using suitable examples. 4

d) Consider the following snapshot of a system: 3+2+3+3+2

	Allocation	*Max*	*Available*
	A B C D	A B C D	A B C D
P0	0 0 1 2	2 0 2 4	1 5 2 0
P1	1 0 0 0	1 7 5 0	
P2	1 3 5 4	2 3 5 6	
P3	0 6 3 2	0 6 5 2	
P4	0 0 1 4	0 6 5 6	

Answer the following Questions:

i. What is the content of the matrix *Need*?

ii. How many comparisons will be necessary to determine whether a system is safe?

iii. Is the system in a safe state? If so, what is the safe sequence? Calculate only the safe sequence according to the safety algorithm.

iv. Calculate the matrix ***Available*** after each calculation.

v. If a request from process P1 arrives for (0,4,2,0) can the request be granted immediately?

8. a) What is the cause of thrashing? Write the basic steps in handling a page fault. 2+7

b) Why is pager instead of swapper used in demand paging? 4

c) Consider the following page reference string: 3x3+3

1, 2, 3, 4, 2, 1, 5, 6, 2, 1, 2, 3, 7, 6, 3, 2, 1, 2, 3, 6

How many page faults would occur for the following replacement algorithms, assuming three frames in the memory for this purpose? What would happen if we could increase the frames in the memory for the same purpose? Assume all frames are initially empty.

i. LRU replacement

ii. FIFO replacement

iii. Optimal replacement