CITY UNIVERSITY OF HONG KONG

Course code & title : CS3103 Operating Systems

Session : Semester B 2005-2006

Time allowed : Two hours

This paper has FIVE pages including this page.

---

1. This paper consists of 9 questions.
2. Answer ALL questions in Section A and any 3 questions in Section B.

Materials and aids permitted for the examination:

Approved Calculator

Section A

Answer ALL questions in this section, each question carries 8 marks.

Question 1

Windows XP classifies interrupts into 32 levels, with 0 being the lowest. The Interrupt Request Level (IRQL) of a processor can be set by OS code so that all interrupts that occur at a level equal to or less than the IRQL of the processor will be blocked.

i) Describe briefly the steps taken by the OS to handle interrupt.
ii) How would the steps be changed if an Interrupt Service Routine (ISR) of level 16 is running and an interrupt of level 20 occurs?
iii) How could the processor IRQL be used to implement interrupt prioritization?

Question 2

Outline the steps in address translation for a processor that use 32 bits logical address in two modes:-

i) segmentation - 10 bits as the segment number and 22 bits as the offset (you may define the structure of the segment descriptor table)
ii) 2-level paging - 10 bits as the 1st level page table directory, 10 bits as the page number and 12 bits as the offset

You may use diagrams to clarify your description.

Question 3

Given the following situation of allocating instances of 3 types of resources A, B and C in a system:

	Allocated			Needed		
	A	B	C	A	B	C
p0	0	1	0	0	0	0
p1	2	0	0	2	0	2
p2	3	0	3	0	0	0
p3	2	1	1	1	0	0
p4	0	0	2	0	0	2

The total number of instances for resources are 7, 2 and 6 respectively for A, B and C. Suppose process p2 makes one additional request for an instance of C, will you grant the resource so that deadlock will not occur? You should show your working steps clearly.

Question 4

Assume you have the following jobs to be executed with one processor :-

Process	Arrival Time	Burst Time
p1	0	8
p2	1	4
p3	2	9
p4	3	5

i) Show with Gantt charts the execution order of these jobs using the non pre-emptive and pre-emptive versions of SJF

ii) Calculate the average waiting time for the algorithms

Question 5

Give a brief explanation of how the following work in disk storage allocation:

i) linked allocation

ii) indexed allocation

For each of the methods, point out its limitations and suggest ways of improvement.

Section B

Answer any THREE questions. ALL questions carry 20 marks.

Question 6

Explain why the following pseudo code may not work:

```
PARBEGIN
   x = x + 1
   x = x + 1
PAREND
```

Name two situations in which the above pseudo code may actually happen. Base on this, distinguish the differences between thread and process.

(8 marks)

Consider the **READER & WRITER problem** which is commonly used to model access to a database. The problem definition is that reading a database is allowed for multiple processes but when one process is writing, no other processes may access. Devise a solution by filling in the missing pseudo code below

Semaphores : mutex, wrt;
Integer : readcount;

```
READER PROCESS                     WRITER PROCESS
WHILE TRUE do                      WHILE TRUE do
Begin                              Begin

 ... read database                  ... write database

End;                               End;
```

To verify your code works, determine for the following cases what semaphores are being waited by which processes? Assume the first process has gained access to the database

a) reader 1, reader 2, writer 1
b) writer 1, reader 1, writer 2, reader 2

(12 marks)

Question 7

Consider the following page reference string, running pure demand paging with 3 available frames:

7 0 1 2 0 3 0 4 2 3 0 3 2

Determine the number of page faults for the following page replacement algorithms by using a diagram to show the usage of individual frames:

i) FIFO
ii) Optimal
iii) LRU (12 marks)

What is Belady's anomaly? Name three factors that affect page faults during program execution, explain how they could be used to avoid thrashing?

(8 marks)

Question 8

The syntax of a blocking file read system call is defined as *n = read(fd, buffer address, bytes),*

where: n is actual of bytes transferred
fd is the file descriptor
buffer address is address for data to be transferred to
bytes is the number of bytes to be read

By tracing the life cycle, identify with brief explanation (preferably in a table form) what would happen in the 4 layers of I/O system when *read( )* is called.

(15 marks)

What is asynchronous and synchronous I/O? Point out where a process may block in the life cycle of *read( )* if the call is made asynchronously and synchronously.

(5 marks)

Question 9

What is a command interpreter? Use the Unix shell as an example to explain the difference between monolithic and kernel approach of OS design.

(6 marks)

What are the differences between kernel mode and user mode, illustrate your answer with the implementation of system call and library call.

(8 marks)

The 2 main functions of OS are interface and arbitration, give 2 example OS components for each of these two purposes, briefly explain.

(6 marks)

*** END ***