COMP 2320 Operating Systems  
1st Semester, 2006-2007  
Mid-term Examination

Time Allowed: 50 Minutes

Answer all SIX questions (total 63 marks)

Questions are printed on both side.

This is a closed book examination

<table>
<thead>
<tr>
<th>Question Number</th>
<th>Full Mark</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>18</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
1.a What are the **FOUR** typical elements in a process image? (2 marks)

- Process Control Block or PCB
- User Program or Program
- User Data or Data
- Stack

1.b What are the **FOUR** typical elements in the Process Control Block? (2 marks)

- Identifiers – like pid, ppid
- Registers
- Stack Pointers
- State Information
(Refer to Table 3.5, Any 4 will do)

1.c In order to manage processes and resources, an operating system usually refers to **FOUR** tables for the current information and status. What are these tables? (2 marks)

- Memory Table
- I/O Table
- File Table
- Process Table
2.a Draw a process state transition diagram with the following states: i) New; ii) Ready; iii) Running; iv) Exit; v) Blocked; vi) Blocked Suspend; vii) Ready Suspend. Please also indicate the interactions between the states. (6 marks)

Figure 3.8(b)

2.b Under what condition will a process be blocked? (1 mark)

Wait for I/O request

2.c What is a typical application that will suspend a running process and examine its content? (1 mark)

debugger
3.a What are the advantages of using Threads instead of using Processes? (2 marks)

Less time to create
Less time to terminate
Less time to switch between two threads
Enhance efficiency in communication (pp. 156-157)

3.b Use a diagram to show a Multithreaded Process Model. (4 marks)
4.a Construct an example such that if mutual exclusion is not enforced in a system. More than one process accessing the same data object at the same time will produce computation/transaction errors. (4 marks)

Account A = $2000
Deposit $1000 to Account A
Read Account A /* A = 2000 */
A = A – 500 /* A = 1500 */
Write $1500 to Account A
A = A + 1000 /* A = 3000 */
Write $3000 to Account A
Withdraw $500 from Account A
Read Account A /* A = 2000 */
A = A – 500 /* A = 1500 */
Write $1500 to Account A

The withdraw transaction is lost because mutual exclusion is not enforced.

4.b Other than Dekker’s software approach and the use of semaphores, name **FOUR** more ways to implement mutual exclusion in a computer system. (2 marks)

Peterson’s algorithm
Test and Set instruction
Exchange Instruction
Special Machine Instructions
Disable interrupts
Monitors
Message Passing
(Any 4 will do)

4.c What are the **FOUR** conditions for Deadlock to happen in a computing system? (4 marks)

1. Mutual exclusion: Only one process may use a resource at a time.
2. Hold-and-wait: A process may hold allocated resources while awaiting assignment of others.
3. No preemption: No resource can be forcibly removed from a process holding it.
4. Circular wait: A closed chain of processes exists, such that each process holds at least one resource needed by the next process in the chain.

(Just the names are OK)
4.d Answer the following questions with the pseudo-code below:

```c
program produce_consumer;
semaphore a = 1;
semaphore b = 0;
semaphore c = 0;
semaphore d = 5;

producer()
{
    for (I = 0; I < 50; I++){
        wait(d);
        wait(a);
        produce
        signal(a);
        signal(c);
    } // end of for loop
    wait(b);
    remove all semaphores
}

c consumer()
{
    for (I = 0; I < 50; I++){
        wait(c);
        wait(a);
        consume;
        signal(a);
        signal(d);
    } // end of for loop
    signal(b);
}

main()
{
    parbegin
    producer();
    consumer();
    parend
}
```

i) What is the purpose of semaphore a? (2 marks)

Guard against critical section or Enforce mutual exclusion

ii) What is the purpose of semaphore b? (2 marks)

The producer will wait until the consumer is done then remove all semaphores or Synchronization between the producer and consumer to remove the semaphores.

iii) What is the purpose of semaphore c? (2 marks)

Synchronization between the producer and consumer (produce first then consume).

iv) What is the purpose of semaphore d? (2 marks)

To enforce a limited buffer of 5 for storing the data.
5.

<table>
<thead>
<tr>
<th>Maximum Claim</th>
<th>Current Allocation</th>
<th>Request (Still Need)</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td>0 1 1 0</td>
<td>0 1 1 0</td>
</tr>
<tr>
<td>P2</td>
<td>2 2 1 1</td>
<td>0 0 1 0</td>
</tr>
<tr>
<td>P3</td>
<td>1 1 2 1</td>
<td>0 1 0 1</td>
</tr>
<tr>
<td>P4</td>
<td>2 1 1 0</td>
<td>1 1 0 0</td>
</tr>
<tr>
<td>P5</td>
<td>0 1 1 1</td>
<td>0 0 0 0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Resource Vector</th>
<th>Available Vector</th>
</tr>
</thead>
<tbody>
<tr>
<td>R1 R2 R3 R4</td>
<td>R1 R2 R3 R4</td>
</tr>
<tr>
<td>2 3 2 1</td>
<td>1 0 0 0</td>
</tr>
</tbody>
</table>

5.a Given the Maximum Claim Matrix, the Current Allocation Matrix, and the Resource Vector, fill-in the Request Matrix, and the Available Vector above. (6 marks)

(1 mark for each correct row)

5.b Use the deadlock detection algorithm to find out whether the above system is deadlock or not. (Please show your steps in deriving your answer) (4 marks)

Q = Available Vector (1 0 0 0)

Mark P5 because it does not have any resource allocated
Mark P1 because it does not need any resource and will finish
Q = (1 1 1 0) after P1 finished
Since Q > P4 (1 0 1 0), mark P4 and update Q to (2 2 1 0)
No more row can be marked, hence P2 and P3 are involved in a deadlock.
6.a  What are the names of the **FOUR** Placement Algorithms used for Dynamic Partitioning.  
(2 marks)

First-Fit  
Best-Fit  
Worst-Fit  
Next-Fit

6.b  For a 32-bit Operating System. If each memory address can hold 1 byte, how much memory can this operating system addressed?  
(1 mark)

\[2^{32} \times 1 \text{ byte} = 4 \text{ Gbyte}\]

6.c  Assuming a paging system using 32-bit for addressing with 16 Kbyte per page. How many bits will be used for the Offset and how many bits will be used for the page number?  
(2 marks)

\[16 \text{ Kbyte} = 2^4 \times 2^{10} = 14 \text{ bit}\]
\[32 \text{ bit} - 14 \text{ bit} = 18 \text{ bit}\]

So we have 18 bit for page number and 14 bit for the offset
6.d Consider the following a snapshot of the main memory for a simple segmentation system. Fill-in the following Base address Registers, and the entries in the Process Segment Tables. (All the addresses listed below are in decimal – base 10).

Base Address Registers:

<table>
<thead>
<tr>
<th>Process A</th>
<th>Process B</th>
<th>Process C</th>
<th>Free-list</th>
</tr>
</thead>
<tbody>
<tr>
<td>1000</td>
<td>1950</td>
<td>3350</td>
<td>3950</td>
</tr>
</tbody>
</table>

Segment Process Table:

<table>
<thead>
<tr>
<th>Seg.#</th>
<th>Process A</th>
<th></th>
<th>Process B</th>
<th></th>
<th>Process C</th>
<th></th>
<th>Free-list</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Address</td>
<td>Length</td>
<td>Address</td>
<td>Length</td>
<td>Address</td>
<td>Length</td>
<td>Address</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>100</td>
<td>0</td>
<td>100</td>
<td>0</td>
<td>1050</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>650</td>
<td>200</td>
<td>1</td>
<td>1500</td>
<td>1</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>850</td>
<td>150</td>
<td>2</td>
<td>2350</td>
<td>2</td>
<td>2650</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td>3</td>
<td>3000</td>
<td></td>
<td>3000</td>
<td></td>
</tr>
</tbody>
</table>

6.e What is the percentage of the memory that is NOT used due to external fragmentation for the above system?

\[
\text{Free memory not due to external fragmentation / total memory} \times 100\% = \frac{750}{4000} \times 100\% = 18.75\%
\]