PES INSTITUTE OF TECHNOLOGY, BANGALORE  
(Autonomous Institute under VTU, Belgaum)  
100 Ft. Ring road, B.S.K. III Stage, Bangalore-560 085  

SEMESTER END EXAMINATION (SEE) B. E. 5TH SEMESTER – DECEMBER, 2009  
CS 302 – OPERATING SYSTEMS

DURATION : 3 HOURS  
Note: ALL QUESTIONS ARE COMPULSORY  
MAX. MARKS : 100

1. a. Give two reasons why caches are useful. What problems do they solve? What problems do they cause? If a cache can be made as large as the device for which it is caching (for instance, a cache as large as a disk), why not make it that large and eliminate the device?  

b. Define the essential properties of the following types of Interactive and real time operating systems  

c. What are the advantages and disadvantages of writing an operating system in a high-level language, such as C?  

d. Would one classify Linux threads as user-level threads or as kernel-level threads? Support your answer with the appropriate arguments.

2. a. Describe supported by a relevant diagram the actions taken by a kernel to context-switch between processes  

b. Can a multithreaded solution using multiple user-level threads achieve better performance on a multiprocessor system than on a single-processor system?  

c. Consider the exponential average formula used to predict the length of the next CPU burst. What are the implications of assigning the following values to the parameters used by the algorithm?  

   i. $\alpha$ (alpha) = 0 and $\tau_0$ = 100 milliseconds  
   ii. $\alpha$ (alpha) = 0.99 and $\tau_0$ = 10 milliseconds  

d. Consider the following set of processes, with the length of the CPU-burst time given in milliseconds  

<table>
<thead>
<tr>
<th>Process</th>
<th>CPU Burst</th>
<th>Arrival Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td>10</td>
<td>13</td>
</tr>
<tr>
<td>P2</td>
<td>12</td>
<td>31</td>
</tr>
<tr>
<td>P3</td>
<td>23</td>
<td>12</td>
</tr>
<tr>
<td>P4</td>
<td>82</td>
<td>09</td>
</tr>
</tbody>
</table>

What is the turnaround time of each process using Preemptive Shortest Job First scheduling criteria.

3. a. Consider a preemptive priority scheduling algorithm based on dynamically changing priorities. Larger priority numbers imply higher priority. When a process is waiting for the CPU (in the ready queue, but not running), its priority changes at a rate $\alpha$ (alpha); when it is running, its priority changes at a rate $\beta$. All processes are given a priority of 0 when they enter the ready queue. The parameters $\alpha$ and $\beta$ can be set to give many different scheduling algorithms.

   i. What is the algorithm that results from $\beta > \alpha > 0$?  
   ii. What is the algorithm that results from $\alpha < \beta < 0$?
Implementing synchronization primitives by disabling interrupts is not appropriate in a processor system if the synchronization primitives are to be used in user-level programs.

Use the wait() and signal() semaphore operations are not executed atomically, then exclusion may be violated.

The method for solving the readers-writers problem without causing starvation involves finding a system consisting of m resources of the same type, being shared by n processes. These can be requested and released by processes only one at a time. Show that the system is lock-free if the following two conditions hold:
- The maximum need of each process is between 1 and m resources
- The sum of all maximum needs is less than m + n

System safe with the following snapshot by applying banker's algorithm

<table>
<thead>
<tr>
<th>Allocation</th>
<th>Max</th>
<th>Available</th>
</tr>
</thead>
<tbody>
<tr>
<td>A B C D</td>
<td>A B C D</td>
<td>A B C D</td>
</tr>
<tr>
<td>P1</td>
<td>0 0 1 2</td>
<td>0 0 1 2</td>
</tr>
<tr>
<td>P2</td>
<td>1 0 0 0</td>
<td>1 7 5 0</td>
</tr>
<tr>
<td>P3</td>
<td>1 3 5 4</td>
<td>2 3 5 6</td>
</tr>
<tr>
<td>P4</td>
<td>0 6 3 2</td>
<td>0 6 5 2</td>
</tr>
<tr>
<td>P5</td>
<td>0 0 1 4</td>
<td>0 6 5 6</td>
</tr>
</tbody>
</table>

A control matrix could be used to determine whether a process can switch from, say, domain B and enjoy the access privileges of domain B. Is this approach equivalent to the access privileges of domain B in those of domain A?

We have a demand-paged memory. The page table is held in registers. It takes 8 to service a page fault if an empty page is available or the replaced page is not modified and 20 milliseconds if the replaced page is modified. Memory access time is 100 nanoseconds. Assume that the page to be replaced is modified 70 percent of the time. What is the acceptable page-fault rate for an effective access time of no more than 200 nanoseconds?

How many page faults occur for your algorithm for the reference string 1 1 2 3 1 2 3 2 5 5 6 8 9 0, in frames using LRU?

We have a system where a file can be deleted and its disk space reclaimed while links to that file remain. What problems may occur if a new file is created in the same storage area or with the same path name? How can these problems be avoided?

A disk drive has 5000 cylinders, numbered 0 to 4999. The drive is currently serving a cylinder 143, and the previous request was at cylinder 125. The queue of pending (FIFO order, is 400, 900, 600, 4324, 4692, 3133, 66, 88, 4977, 11. What is the total number of cylinders that the disk arm moves to satisfy all the pending requests if you apply a scheduling algorithm?

Note on VxWorks?