|  |  |  |
| --- | --- | --- |
| **KING SAUD UNIVERSITY**  **COLLEGE OF COMPUTER AND INFORMATION SCIENCES Computer Science Department** | | |
| **CSC 227: Operating System** | **Tutorial# 5**  **Due: Sun, March 26 (12-1)** | **2nd Semester 1437-1438**  **Spring 2017** |

**Question#1:**

*Select (T) for true or (F) for False.*

|  |  |  |
| --- | --- | --- |
| True/False | | Statement |
| **T** | F | 1. If the child process exceeded the memory allocated for it, the parent process may terminate it. |
| T | **F** | 1. TerminateProcess() is sent by a process that completes its execution. |
| T | **F** | 1. A child process which terminated before returning its exit status to the parent is an orphan process. |
| T | **F** | 1. In distributed system environment, shared memory IPC model provides fast communication. |
| **T** | F | 1. In producer-consumer concept a buffer is used in order to enable the producer and consumer to run concurrently. |
| **T** | F | 1. Message acknowledgment is used by the message passing IPC model. |
| T | **F** | 1. If a hardware platform does not have a timer, the preemptive scheduling is the best choice. |
| **T** | F | 1. I/O device speed can affect the turnaround time of a process. |
| T | **F** | 1. Waiting time is the time spent by a process in the I/O device queue. |

**Question#2:**

*Circle ALL correct answers. Note that there might be multiple correct answers; also, there might be no correct answer.*

1. In Unix, a parent process may terminate a child using:
   1. **abort()**
   2. exit()
   3. TerminateProcess()
2. Which process operation can help on detecting security threats:
   1. Destroy(kill)
   2. **suspend**
   3. abort
3. Which state is helpful on managing priorities:
   1. wait
   2. **blocked-wait**
   3. ready
4. In the shared-memory IPC model:
   1. A processor searches for a variable in remote caches.
   2. **A processor searches for a variable in its own cache.**
   3. **A processor searches for a variable in main memory.**
5. in the message passing IPC model, if CPU1 cannot find a variable in its memory, CPU1 sends the message:
   1. send( CPU1, message)
   2. **send(message)**
   3. send()
6. In producer-consumer concept, which condition indicates that the buffer is empty:
   1. ((out + 1) % BufferSize) = in
   2. The in pointer precedes the out pointer
   3. The out pointer precedes the in pointer
7. In non-preemptive scheduling the process continues running until:
   1. **It terminates.**
   2. **It switches to waiting.**
   3. It is replaced by another process that meets the scheduling algorithm criteria.
8. Which of the following scheduling algorithms metrics should be maximized:
   1. **Throughput.**
   2. Response time.
   3. **CPU utilization.**
   4. Variance.
9. Which of the following scheduling algorithms metrics are used in time-sharing environment:
   1. Throughput.
   2. Waiting time.
   3. **Response time.**
   4. **Variance.**

**Question #3:**

Given a bounded buffer of size =8. Answer the following questions:

1. What is the condition that must be satisfied to have an empty buffer? when in = out.
2. What is the condition that must be satisfied to have a full buffer? ((in +1) % BUFFER\_SIZE) = out.
3. Complete the table below that updates the values of both in and out pointers.
4. Draw the buffer after applying the six actions in the table.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| *SR* | *Action* | *in* | *out* | *Empty?* | *Full?* | *Comment* |
| *0* | *Initial Position* | 0 | 0 | Yes | No |  |
| *1* | *Produce 3 items* | 3 | 0 | No | No |  |
| *2* | *Consume 1 item* | 3 | 1 | No | No |  |
| *3* | *Consume 2 items* | 3 | 3 | Yes | No |  |
| *4* | *Produce 4 items* | 7 | 3 | No | No |  |
| *5* | *Produce 6 items* | 2 | 3 | No | Yes |  |
| *6* | *Consume 2 items* | 2 | 5 | No | No |  |

*Draw the final status of the buffer:*

***In =2, out = 5***