**Studytonight – OS test 1 – Aditya Jain**

1. **System calls are usually invoked by using:**
2. **A software interrupt**
3. Polling
4. An indirect jump
5. A privileged instruction

Soln: Privileged instruction cannot be the answer as system calls is done from user mode and privileged instruction cannot be done from the user mode.

1. **A processor needs software interrupt to**
2. Test the interrupt system of the processor
3. Implement co-routines
4. **Obtain system services which need execution of privileged instructions**
5. Return from subroutine
6. **A CPU has two modes-privileged and non-privileged. In order to change the mode from privileged to non-privileged:**
7. A hardware interrupt is needed
8. **A software interrupt is needed**
9. A privileged instruction (which does not generate an interrupt) is needed
10. A non-privileged instruction (which does not generate an interrupt) is needed
11. **Which of the following is an example of a spooled device?**
12. The terminal used to enter the input data for the C program being executed
13. **An output device used to print the output of a number of jobs**
14. The secondary memory device in a virtual storage system
15. The swapping area on a disk used by the swapper

Soln: spooling (simultaneous peripheral operations online) is a technique in which an intermediate device such as disk is interposed between process and low speed i/o device. For ex: in printer if a process attempt to print a document but printer is busy printing another document, the process, instead of waiting for printer to become available, write its output to disk. When the printer become available the data on disk is printed. Spooling allows process to request operation from peripheral device without requiring that the device be ready to service the request.

1. **Which scheduling policy is most suitable for a time-shared operating system?**
2. Shortest Job First
3. **Round Robin**
4. Elevator
5. First come First Serve
6. **Which of the following actions is/are typically not performed by the operating system when switching context from process A to process B?**
7. Saving current register values and restoring saved register values from process B
8. Changing address translation tables
9. **Swapping out the memory image of process A to the disk**
10. Invalidating the translation look-aside buffer

Soln: processes are generally swapped out from memory to disk when they are decided to be suspended.

1. **A multi-user, multi-processing operating system cannot be implemented on hardware that does not support: (more than one correct answers)**
   * + - 1. **Address translation**
         2. DMA for disk transfer
         3. **At least two modes of CPU execution (privileged and non-privileged)**
         4. Demand paging

Soln: Address translation is needed to provide memory protection so that a given process does not interfere with another.

We also need at least 2 modes of execution to ensure user processes share resources properly and OS maintains control. This is not required for a single user OS like early version of MS-DOS.

Demand paging and DMA enhances the performances- not a strict necessity.

1. **Consider a set of n tasks with known runtimes R1, R2, R3, ….Rn to be run on a uniprocessor machine. Which of the following processor scheduling algorithms will result in the maximum throughput?**
2. Round-Robin
3. **Shortest Job First**
4. Highest Response Ratio Next
5. First Come First Served
6. **Consider the following statements about user level threads and kernel level threads. Which one of the following statements is FALSE?**
   1. Context switch time is longer for kernel level threads than for the user level threads
   2. User level threads do not need hardware support
   3. Related kernel level threads can be scheduled on different processors in a multi-processor system
   4. **Blocking one kernel level thread blocks all related threads.**

**Soln: kernel threads do not get blocked for I/O or system call, when one thread blocks.**

1. **Which of the following does not interrupt a running process?**
   1. A device
   2. Timer
   3. **Scheduler process**
   4. Power failure

Soln: scheduler process is meant to decide which ready process next should run on CPU

1. **A CPU generally handles an interrupt by executing an interrupt service routine**
   1. As soon as an interrupt is raised
   2. By checking the interrupt register at the end of fetch cycle
   3. **By checking the interrupt register after finishing the execution of the current instruction**
   4. By checking the interrupt register at fixed time intervals

Soln:Hardware detects interrupt immediately, but CPU acts only after its current instruction. This is followed to ensure integrity of instructions.

1. **Which of the following is/are true of the auto-increment addressing mode?**
   * 1. It is useful in creating self-relocating code
     2. If it is included in an Instruction Set Architecture, then an additional ALU is required for effective address calculation
     3. The amount of increment depends on the size of the data item accessed
        + 1. i only
          2. ii only
          3. **iii only**
          4. ii and iii only

soln: different datatypes have different memory requirements. So the amount of increment depends on the datatype of the data item. (1 byte for char, 4 bytes for int and so on…)

1. **Which of the following statements are true?**
   * 1. Shortest remaining time first scheduling may cause starvation
     2. Preemptive scheduling may cause starvation
     3. Round robin is better than FCFS in terms of response time.
        + 1. i only
          2. i and ii only
          3. ii and iii only
          4. **i, ii and iii**

soln: all 3 claims are true.

1. A thread is usually defined as a “light weight process” because an operating system (OS) maintains smaller data structures for a thread than for a process. In relation to this, which of the following is TRUE?
   1. On per-thread basis, the OS maintains only CPU register state
   2. The OS does not maintain a separate stack for each thread
   3. **On per-thread basis, the OS does not maintain virtual memory state.**
   4. On per-thread basis, the OS maintains only scheduling and accounting information.

Soln: Threads share address space of Process. Virtually memory is concerned with processes not with Threads.

1. **Let the time taken to switch between user and kernel modes of execution be t1 while the time taken to switch between two processes be t2. Which of the following is TURE?**
   1. t1>t2
   2. t1=t2
   3. **t1<t2**
   4. nothing can be said about the relation between t1 and t2

soln: since process switching also involves mode switching, process switching time is more time consuming than mode switching time