**SOLUTION**

Fall 2020 CSC 332 Quiz 2 (Ch I) 300 Points 75 minutes

Each question is 100 Points.

Q1.

Consider process management with RR as discussed in Sec I.2.

Suppose we did the same shell command that created the 4 processes for ./cpu

Make these assumptions:

1. There is only one cpu.
2. Initially all the 4 processes are in ready list in the order A, B, C,D.
3. There are only 5 processes in the system—A, B, C, D, and the shell.
4. **All processes are entirely in physical memory.**
5. The only interrupts in the system are the Hardware Timer and interrupts caused by this program.
6. The cpu scheduler strictly follows the RR algorithm, with **time slice = 10 seconds**
7. **The spin(1) will take exactly 1 second of computation time. Any other computation by the cpu will take zero time—ex. the initial if statement, checking the loop condition, service routine execution, etc. will take exactly zero time.**
8. **When a process executes the printf statement in cpu.c, it does NOT get blocked; its output is stored in memory and the process continues its execution.** The OS sends various outputs to the printer, in FIFO order, from time to time when the printer becomes idle. The print operation for one printf output takes 1 second. The output device prints things in strictly First-come-first-served order.

What is the sequence of the first three letters printed?

Write down **all such possible sequences** of the first 3 letters.

Explain your answer in less than 100 words.

**Answer:**

The only possible sequence of first three letters is: A, A, A

**Explanation:**

Time instant 0: Proc A start execution.

Time interval 0-1: proc A does spin.

Time instant 1: proc A makes request to print A. The service routine for this system call copies A into a buffer and starts print operation on device. Proc A resumes execution.

The above sequence of events at proc A continues for at least time instant 4.

So output printed will have A, A, A first.

Other processes don’t become Running during time 0-4 because proc A does not get blocked and time slice is 10.

**Q2.**

Consider a paging scheme.

Suppose the page frame size is 16 bytes.

Suppose all of process A is currently loaded in memory.

Assume the page i of process A is loaded in **page frame**

**5i** (for any page i of the process).

In process A, there is an array B[0..100].

Assume that **each element of B occupies 2 bytes.**

**The logical address** of the beginning byte of B[50]

(i.e., the byte 0 of B[50]) **is 1200.**

Compute the **physical address of beginning byte of B[3].**

Show your calculations. **Clearly write down the final answer in one place.**

**Answer:**

**For beginning byte of B[3]:**

**Logical addr= 1200 – (50-3) \* 2=1200- 47\*2 = 1200 – 94 = 1106**

**Page number= 1106/16 = 69**

**Offset=2**

**Page frame number=5\*69= 345**

**Physical address= 345\*16 + 2= 5522 (ANSWER).**

**Q3.**

(a) Can a process switch state from Ready state to Blocked state? Explain.

**Answer**: No.In order to get into blocked state, it has to make an explicit request for something. But in order to make an explicit request, it has to have the cpu. In ready state it does not have the cpu.

(b) **True or False?** **(Explain):** In dynamic partitioning, once a process begins its execution first time, and its BASE and BOUND registers have been set, it will have the same BASE and BOUND register values each time it executes (say after being blocked, etc.).

**Answer:** No. If a degragmentation is done, and this process gets moved to a new location, then its BASE value will change.