

Shadow Homework 2 – System Programming (Processes, Threads, Synchronization and Scheduling)

Posted February 15, 2007 (NOT GRADED)

Shadow Homework 2 on Processes, Threads, Synchronization and Scheduling

1. Draw a state transition diagram with three process states “Ready” “Running” and “Blocked”. Indicate allowed transitions between these states .
2. Give at least two differences and two similarities between an application that is implemented using multiple processes and one that is implemented using multiple threads. Your answer should emphasize the performance and resource issues in each case.
3. List and explain four resources that must be saved and restored during a process context-switch.
4. Explain what is the problem with this implementation of the one-writer many-readers problem?
(Note, “Up” is equivalent to “sem_post” & “Down” is equivalent to “sem_wait”)

```

int readcount;          // shared and initialized to 0
Semaphore mutex, wrt;  // shared and initialized to 1;

// Writer :
Down(wrt);
/* Writing performed*/
Up(wrt);

// Readers :
Down(mutex);
readcount := readcount + 1;
if readcount == 1 then Down(wrt);
Up(mutex);
/*reading performed*/
Down(mutex);
readcount := readcount - 1;
if readcount == 0 then Up(wrt);
Up(mutex);

```

5. Consider the following table about processes:

Process	Priority*	Burst Time	Arrival Time
P ₁	4	100ms	0ms
P ₂	2	20ms	30ms
P ₃	1	50ms	40ms
P ₄	3	5ms	60ms

* A smaller priority number means a higher priority process

The schedule for First Come First Served is shown below

FCFS: P₁ P₂ P₃ P₄
 |-----|-----|-----|-----|

0 100 120 170 175

a) Explain the term Average Waiting Time.

c) What is the most important characteristic of Shortest Job First scheduling?

d) Edit the following skeleton schedules for the process table to show the scheduling disciplines:

i) Preemptive Shortest Job First

|-----|-----|-----|-----|

ii) Preemptive Priority

|-----|-----|-----|-----|

iii) Round Robin (with quantum 40ms)

|-----|-----|-----|-----|