

# CS 385: HW4 – Due April 22, 2004

Jon A. Solworth

April 16, 2004

1. **Banker's Algorithm** Given the following configuration, where each pair *allocated/need* refers to the number of resources currently allocated and needed for the process to complete.

(a) Is the following configuration *safe*?

	R1	R2	R3
Total	5	6	3
$p_0$	1/2	2/1	1/1
$p_1$	2/0	1/2	0/1
$p_2$	2/1	2/0	1/1

(b) Is the configuration obtained by adding one more unit of R1 (total of 6) to the above configuration *safe*?

2. **Starvation** In each of the following cases, clearly state if there exists a possibility for starvation and justify your answer (please be precise and to the point).

(a) There exists 10 levels of priority in the running queue. A process is assigned an initial priority at creation. The process may move up to 4 levels of priority, from the initially assigned value, if all other processes are above its current level of priority. Similarly, a process may move down up to 2 levels of priority, from the initially assigned value, if it is the only process at its current priority level and there are no processes in any of the levels above it.

(b) In a queue for the list of processes waiting for a semaphore, the processes are woken up based on their process id's.

3. **Process States** What happens in terms of process states for the parent and the child processes for the following events (clearly state what transition(s) happen along with the starting and ending process states).

(a) When the parent process executes a `fork()` system call.

(b) When the child process executes an `exit()` system call.

- (c) Assume there exists a pipe between the parent and the child process, and the pipe fd's are setup such that the child is the writer and the parent is the reader. Discuss the transitions when the parent tries to read from the pipe in the case when (i) there is data to be read and (ii) the pipe is empty, but the write end is not closed by the child.

4. **Scheduling Algorithms** Consider the following set of processes, with the length of CPU times needed for completion.

Process	Time
$p_1$	10
$p_2$	1
$p_3$	2
$p_4$	4
$p_5$	6

The processes are assumed to have arrived in the order  $p_1, p_2, p_3, p_4, p_5$  all at time 0. Consider the scheduling algorithms (i) First Come First Served, (ii) Shortest Job First and (iii) Round Robin (with time slice = 2 units) when answering the following questions.

- (a) What is the *turnaround time* for each process for each of the above scheduling algorithms?
- (b) What is the *waiting time* for each process for each of the above scheduling algorithms?
- (c) Which is the scheduling algorithm with the least average waiting time?
5. **Little's Formula** Assume that a supermarket doesn't want more than 8 customers waiting on a queue at any time. If the arrival rate is 2 customers/minute what is the waiting time ?
6. **Threads** What are the advantages and disadvantages of threads over processes. (As far as possible state your answer as a series of bullet points than just one big paragraph).