UCLA CS 111 Winter 2009
Practice Midterm

I Processes

In the following questions, circle a *minimal* set of system calls that can implement the following command lines in a shell implemented like Lab 1's ospsh. This shell *does not* support advanced features like job control or after. Don't include system calls required only for error reporting or cleaning up zombie processes. If you think you might have circled more system calls than we expect, write down why.

3 *command* &

A. read D. fork G. signal/sigaction J. open
B. write E. dup2 H. exit K. close
C. pipe F. wait/waitpid I. execvp L. chdir

4 *command1* | *command2*

A. read D. fork G. signal/sigaction J. open
B. write E. dup2 H. exit K. close
C. pipe F. wait/waitpid I. execvp L. chdir

5 *command* <*file*

A. read D. fork G. signal/sigaction J. open
B. write E. dup2 H. exit K. close
C. pipe F. wait/waitpid I. execvp L. chdir

6 *command1* && *command2*

A. read D. fork G. signal/sigaction J. open
B. write E. dup2 H. exit K. close
C. pipe F. wait/waitpid I. execvp L. chdir

7 cd *dir1* || cd *dir2*

A. read D. fork G. signal/sigaction J. open
B. write E. dup2 H. exit K. close
C. pipe F. wait/waitpid I. execvp L. chdir

II Scheduling

Optimizatron decides to design a variant of Shortest-Job-First scheduling that will have slightly worse average wait time, but eliminate starvation. His idea is this:

- Use a preemptive version of strict priority scheduling.
- When a request arrives, set its priority value to its amount of work. (We assume work amounts are integers >= 1.)
- Use *priority aging* to give requests higher priority the longer they wait. Specifically, every 5 time units that a job waits, reduce the job's priority value by 1, down to a minimum of 1.

His friend Bert offers to implement this strategy. Unfortunately, Bertha's implementation still suffers from starvation!

a)What did Bert do wrong? Is this a fundamental problem with Optimizatron's design, or is there a different implementation that would meet his goals?

The following requests arrive at a scheduler at the times indicated. The scheduler begins to execute requests at time 0. Assume that the context switch overhead is also 0.

Request	**Arrival time**	**Amount of work**
A	−2	1
B	−1	3
C	**???**	**???**

b) Running these jobs with two different cooperative schedulers gives the following average metrics.

Scheduler X Average waiting time = 4
Scheduler Y Average turnaround time = 8

Figure out schedulers and characteristics of C that produce these results. Note that C's characteristics must be the same under both Scheduler X and Scheduler Y. You may assume that C's arrival time and amount of work are integers, and C arrives at a different time than A or B. *Hints:* This restricts the possible scheduling orders, making your work easier. One of the equations you'll generate has only one variable in it: we'd advise solving that equation first.

– **Scheduler X**:
– **Scheduler Y**:
– **C's arrival time**:
– **C's arrival time**:
– **C's amount of work** :

c)List one advantage and one disadvantage of preemptive round-robin scheduling relative to other scheduling algorithms, in terms of metrics such as average waiting time, average turnaround time, possibility of starvation, and average utilization.

III Synchronization

You have been hired by NASA to study Operating Systems issues in their latest Mars probe. Consider the following situations:

a) The probe has several interconnected fuel tanks. The tanks are connected in a ring and fuel can be pumped between adjacent tanks by any of the processes running on the probe. To avoid race conditions, processes use the following algorithm for moving fuel: lock the tank from which fuel is being moved; lock the tank to which fuel is being moved; move the fuel between tanks (increment the contents of one and decrement the contents of the other); release the locks in the reverse order. What is the most serious problem this solution has?Give an example illustrating the problem. Suggest a way to fix it.

b) This problem can be reduced to another problem that was discussed in class. Name the problem, and show how the two problems are equivalent (4 pts).

c) Because we can reduce the fuel problem to the class problem (the one discussed in class), we can use the known solution for that problem. Other than avoiding the problem you identified above, name an advantage that using the known solution has. (4 pts)

d) This probe is a self-contained system being sent far from its controllers into a dangerous and unpredictable environment with the intent of reporting interesting data back to its owners. Discuss 3 ways in which the **operating system** of this probe must differ from the operating system of a general purpose earthbound computer.