Operating Systems, Spring 2015

Midterm

2:10pm $\sim$ 3:50pm, Tuesday, April 21, 2015

INSTRUCTIONS:

1. This is a *closed-book* exam.
2. Try to solve all of the problems.
3. Try to give short answers. (Hint: An answer need not always be longer than the question.)
4. No cheating.
5. Please hand in both the exam sheet and the answer sheet.
6. Please note that unless otherwise stated, all the line numbers for the program listings are for reference only.

1. (10%) Measurements of a certain system have shown that the average process runs for a time T before blocking on I/O. A process switch requires a time S, which is effectively wasted (overhead). For round-robin scheduling with quantum Q, give a formula for the CPU efficiency (i.e., the useful CPU time divided by the total CPU time) for each of the following:

   (a) $S < Q < T$

   (b) $Q = S$

   To simplify the answers, you may assume Q divides T evenly.

2. (20%) Suppose that two processes, P_1 and P_2, are running in a uniprocessor system. P_1 has two threads. P_2 has three threads. All threads in both processes are CPU-intensive; that is, they never block for I/O. The operating system uses simple round-robin scheduling.

   (a) Suppose that all of the threads are user-level threads, and that user-level threads are implemented using a single kernel thread per process. What percentage of the processor's time will be spent running P_1's threads?

   (b) Suppose instead that all of the threads are kernel threads. What percentage of the processor's time will be spent running P_1's threads?

3. (20%) Consider the interprocess-communication scheme where mailboxes are used. Suppose a process P wants to wait for two messages, one from mailbox A and one from mailbox B. What sequence of `send` and `receive` should it execute so that the messages can be received in any order?

4. (10%) Disk requests come in to the disk driver for cylinders 10, 22, 20, 2, 40, 6, and 38, in that order. A seek takes 5 msec per cylinder moved. How much seek time is needed for

   (a) Closest cylinder next, and

   (b) Elevator algorithm (initially moving upward).

   In all cases, the arm is initially at cylinder 20.

5. (10%) A computer has six tape drives, with n processes competing for them. Each process may need two drives. For which values of n is the system deadlock free?

6. (10%) The banker's algorithm is being run in a system with m resource classes and n processes. In the limit of large m and n, the number of operations that must be performed to check a state for safety is proportional to $m^a n^b$. What are the values of a and b?

7. (20%) What would be the output of the following C program that uses the Pthreads API? (*Note that the line numbers are for references only.*)

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <unistd.h>
4  #include <pthread.h>
5  #include <sys/types.h>
6  #include <sys/wait.h>
7
8  static void *runner(void *param)
9  {
10     ++(* (int*) param);
11     pthread_exit(0);
12 }
13
14 int main(int argc, char **argv)
15 {
16     int status;
17     int value = 100;
18     pid_t pid = fork();
19     if (pid > 0) {
20         waitpid(-1, &status, 0);
21         printf("A = %d\n", ++value);
22     }
23     else if (pid == 0) {
24         pid_t pid = fork();
25         if (pid > 0) {
26             waitpid(-1, &status, 0);
27             printf("B = %d\n", value++);
28         }
29         else if (pid == 0) {
30             pid_t pid = fork();
31             pthread_t tid;
32             pthread_create(&tid, NULL, runner, &value);
33             pthread_join(tid, NULL);
34             if (pid > 0) {
35                 waitpid(-1, &status, 0);
36                 printf("C = %d\n", --value);
37             }
38             else {
39                 printf("D = %d\n", value--);
40             }
41         }
42         else {
43             return 1;
44         }
45     }
46     else {
47         return 1;
48     }
49     return 0;
50 }
```