CSE 2431 Homework 1

Instructor: Adam C. Champion, Ph.D.
Due: Thursday, January 30, 2020 (at start of class) (100 points)

Please type your homework. Handwriting is not acceptable.

Listing 1: Code for Problem I

```c
#include <...> /* Assume all necessary headers are included */

pid_t pid;

int value = 8;

int main() {
    pid = fork();
    
    if (pid > 0) {
        value *= 2;
        printf("pid=%d, value=%d\n", pid, value);
    }

    printf("pid=%d, value=%d\n", pid, value);

    exit();
}
```

I. Processes (30 points)

(a) What will be the output of the child process? (Assume that the child process’s pid is 1800). (10 points)

(b) What will be the output of the parent process? (10 points)

(c) Briefly describe each of the following system calls in your own words. You may need to consult outside sources to answer these questions; please cite all sources you use (e.g., Linux man pages). (10 points)
   - getpid() (3 points)
   - kill() (3 points)
   - execve() (4 points)

II. Threads (30 points)

(a) What resources are private to each thread in a multithreaded process? (10 points)

(b) Describe two differences between user-level and kernel-level threads. In what circumstances is one better than the other? (10 points)

(c) What are scheduler activations? What are their advantages and disadvantages? (10 points)
III. CPU Scheduling (40 points)

<table>
<thead>
<tr>
<th>Process</th>
<th>Burst Time</th>
<th>Priority</th>
<th>Arrival Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td>10</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>P2</td>
<td>9</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>P3</td>
<td>2</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>P4</td>
<td>14</td>
<td>5</td>
<td>6</td>
</tr>
<tr>
<td>P5</td>
<td>6</td>
<td>4</td>
<td>8</td>
</tr>
</tbody>
</table>

Table 1: Process Information.

Consider the processes described in Table 1. Higher priority numbers indicate higher-priority processes (e.g., priority 5 is higher than priority 1).

Questions: What is the average wait time of these processes for each of the following scheduling algorithms? (Draw a Gantt chart for each algorithm.)

(a) First Come First Serve (FCFS)
(b) Non-preemptive Shortest Job First (NP-SJF)
(c) Preemptive Shortest Job First (P-SJF)
(d) Preemptive Priority Scheduling
(e) Round Robin, with the following assumptions:

Assumption (1). The scheduling time quantum is 5 time units.

Assumption (2). If a new process arrives at the same time as the time slice of the executing process expires, the OS puts the executing process at the end of ready queue after the new process.