Homework 1 for CS153 (Fall 2018)

Due: Wednesday 10/17, 2018

Instructions:

* Be brief. You will be graded for correctness, not on the length of your answers.
* Make sure to write legibly. Incomprehensible writing will be assumed to be incorrect.

1. OS is a sleeping beauty. It can be woken up by four kinds of events. What are these events? Give an example for each kind of events. (16 points)

2. Explain what happens in the computer when you press a key on the keyboard (Hint: read slides about how the computer receives a packet from Ethernet). (16 points)

3. What is a process? More specifically, what does a process contain? (16 points)

4. There is a sleep system call In UNIX. It pauses the current process (who makes this system call) for a specified number of seconds. Explain what happens in the OS kernel after this sleep system call is invoked. (Hint: focus on process state transitions, wait queues, scheduling). (16 points)

5. Run “sleep 3” and “exec sleep 3” in your shell respectively. Describe what happens, and explain why it happens this way. (Hint: think about how “fork” and “exec” work, check slide No.13 in lec04.pdf for how a shell works.) (16 points)

6. Consider the following program:

```
1 int main() {
2 int count = 0;
3 int pid;
4
5 if( !(pid = fork()) ) {
6      while((count < 2) && (pid = fork()) ) {
7          count++;
8          printf("%d", count)
9      }
10     if (count > 0) {
11         printf("%d", count);
12     }
13 }
14 if(pid) {
15     waitpid(pid, NULL, 0);
16     count = count << 1;
17     printf("%d", count)
18     }
19 }
```

Some explanation of the syntax:

- In C, when we have a condition such as that in line 6, we evaluate the first term and only if it is true, do we execute the second term. If it is false, we do not evaluate the second term.
- (pid = fork()) in the same line executes the fork, assigns the return value to pid and uses that return value to evaluate the condition.
- Count = count << 1 is a logical left shift by 1 bit. It’s a cheap way to multiply by 2 for integers

A. How many processes are created during the execution of this program? Explain. (10 points)

B. List all the possible outputs of the program. (10 + 3 bonus points)