

## Problem Session 1: Processes and Scheduling

Wednesday, April 1, 2020

1. How many lines of output does the following function print? Give your answer as a function of  $n$ .

```
void foo(unsigned n){
    for(unsigned i = 0; i < n; i++){
        fork();
    }
    printf("hello\n");
}
```

2. Consider the following program:

```
void f(){
    printf("2");
}

int main(){
    int check = 0;

    if(fork() == 0){
        check = 1;
    }

    if(fork() == 0){
        printf("0");
    } else {
        printf("1");
    }

    if(check){
        f();
    }
    exit(0);
}
```

Which of the following outputs are possible:

- (a) 112002
- (b) 211020
- (c) 102120
- (d) 122001
- (e) 100212

3. What are the possible output sequences from the following program?

```
int main(){
    if(fork() == 0){
        printf("a");
        exit(0);
    } else {
        printf("b");
        wait();
    }
    printf("c");
    exit(0);
}
```

4. Given the following jobs, compute the latency and response time for each job, along with the average response time, for FIFO, STCF, and RR scheduling algorithms. Assume a time slice of 10 for RR.

Job	Length	Arrival Time	FIFO		STCF		RR	
			Latency	Response	Latency	Response	Latency	Response
0	85	0						
1	30	10						
2	35	15						
3	20	80						
4	50	85						
Average:								

5. Consider a set of three jobs, A, B, and C, running concurrently on a computer system:

- Job A arrives first at time 0 and uses the CPU for 50ms before finishing.
- Job B arrives at time 1. Job B loops five times; for each iteration of the loop, B uses the CPU for 2ms and then does I/O for 8ms.
- Job C arrives at time 2. Job C is identical to Job B except for the arrival time.

Assuming there is no overhead to doing a context switch, identify when A, B, and C will finish for each of the following scheduling algorithms:

- RR with a 1ms time slice
- RR with a 20 ms time slice
- Multilevel feedback queue with four levels with a time slice of 10 in the highest priority queue, 20 in the next, 40 in the next, and 80 in the lowest priority queue. Priorities reset every 200ms.