

Exam #1

CSE 3320.002

Fall 2014

Name: \_\_\_\_\_

UTA ID: \_\_\_\_\_

“I certify that the following work is my work alone and I will follow the highest standards of integrity and uphold the spirit of the Honor Code”

Signature: \_\_\_\_\_

Directions: This is a closed book, closed notes exam. You may use a hand written 3x5 note card with notes. Please answer the questions briefly. Complete sentences are not necessary. Write your answers legibly. Unreadable answers will be counted wrong. You may write on back if needed.

1. [5 pts] How do system calls vary from function calls in terms of execution? Why do they vary?
2. [5pts] Explain the difference between cooperative and pre-emptive multitasking. What is the advantage of each method?

3. [5pts] For the following code, describe what happens, in what order, and what will get printed (in what order). For return values from fork use any integer that makes sense from a system standpoint:

```
pid_t pid_1;
pid_t pid_2;
int status;

pid_1 = fork();

if (!pid_1)
{
    wait(&status);
    printf("A, my PID is %d\n", pid_1 );
    if (!(pid_2 = fork()))
    {
        wait(&status);
        printf("B, my PID is %d\n", pid_2 );
    }
    else
    {
        wait(&status);
        printf("C, my PID is %d\n", pid_2 );
    }
    exit( 0 );
}
wait(&status);
printf( "D, my PID is %d \n", pid_1 );
exit( 0 );
```

4. [5pts] For the following code, describe what happens, in what order, and what will get printed (in what order). For return values from fork use any integer that makes sense from a system standpoint:

```
execl( "/bin/mkdir", "/bin/mkdir", "foo", NULL );

printf("I am in the parent \n");

child_pid = fork();
if (child_pid == 0)
{
    printf("I am in the child \n")
}
else
{
    int status;
    wait( &status );
    printf("Leaving the parent \n");
    exit( 0 );
}
```

5. [10pts] For a CP/M disk, the disk had 40 tracks of 32 sectors each sector was 512 bytes long and the disk had two sides. The disk had the “standard” CP/M disk directory.

- A. What is the raw disk capacity. You do not need to calculate the final result - just show the calculation, and your reasoning.
- B. For this disk, how many 1 byte files could be stored?
- C. What is the largest single file possible? You do not need to calculate the final result - just show the calculation, and your reasoning.
- D. For the large file in 8C , called “file.txt”, describe the contents of the first two directory entries.

6. [5pts] Draw the Mac OS System 1 memory layout

7. [5pts] In the application partition of Mac OS System 1:
- A. How was the heap guarded? Where there problems with this method?
  - B. How was the stack guarded? Where there problems with this method?

8. [5pts] Explain:
- A. What is the Process Control Block? What is its purpose?
  - B. How many PCBs are there?
  - C. Name three items in the PCB.

9. [10pts] What technique did PALM OS use to correct memory fragmentation? How did they avoid causing problems for running programs? Describe it and draw it.



10. [45 pts]

The following is a list of processes

Process ID	Arrival Time	Runtime (seconds)	Priority
1	0	15	4
2	2	3	2
3	3	10	3
4	3	6	1
5	9	3	1
6	12	1	4
7	12	8	5
8	20	3	2

Where arrival time and running time are in seconds, and memory size in K Bytes. If appropriate the time quantum = 1 second. Lower priority values are higher priority. When arrival time is the same, select lowest process number first, if more than one process can be selected use "most fair" - allow next (newest) process to run next. Please do not "combine" scheduling policies, unless scheduler normally does that.

1. Please show the GANTT chart for a FCFS scheduler.
  1. What is the turnaround time for Job 2 and Job 7?

2. Please show the GANTT chart for a Shortest Job Next (SJN) scheduler (with no preemption).
  1. What is the response time for Job 2 and Job 6?

Same table:

Process ID	Arrival Time	Runtime (seconds)	Priority
1	0	15	4
2	2	3	2
3	3	10	3
4	3	6	1
5	9	3	1
6	12	1	4
7	12	8	5
8	20	3	2

3. Please show the GANTT chart for a SJN scheduler, with preemption.

Same table:

Process ID	Arrival Time	Runtime (seconds)	Priority
1	0	15	4
2	2	3	2
3	3	10	3
4	3	6	1
5	9	3	1
6	12	1	4
7	12	8	5
8	20	3	2

4. Please show the GANTT chart for a Priority scheduler, with no preemption.

5. Please show the GANTT chart for a Priority scheduler, with preemption.

6. Comparing: 1. FCFS, 2. SJN with preemption, 3. Priority 4. Priority with preemption
  1. Which scheduler has the minimum turnaround time for Jobs 4 and 5?

2. Which scheduler has the worst turnaround time for Jobs 4 and 5?

3. Which scheduler has the best average response time?

Bonus Question (5pts)

Due to the, seemingly, fragile code of the parser in the bash shell it has taken 3 patches to patch the bash shell vulnerability we discussed on Wednesday. It may still not be 100% fixed. Potentially, it could be weeks before all exploit paths are closed. At this point what would you recommend?

1. Continue to try and correct each new failure case and issue patch after patch to the old code as problems are found.
2. Get rid of the fragile parser and rewrite it from scratch cleanly.

Justify your answer. I want your reasoned opinion as a software developer. Either option can be correct if you give good reasons.