Midterm Exam Results for Mualla Argin

Score for this quiz: **28.04** out of 50
Submitted Mar 10 at 4:44pm
This attempt took 50 minutes.

Question 1

1 / 1 pts

First Come First Served (FCFS) scheduling algorithm performance can never equal the performance of Shortest Job First (SJF) for average response time.

- True
- False

Correct!

Question 2

1 / 1 pts

The Process Control Block (PCB) does not contain any scheduling information about a process.

- True
- False

Correct!

Question 3

1 / 1 pts

In typical I/O interrupt schemes, the CPU checks for Interrupts after executing every instruction.

Correct!

- True
- False

Question 4

1 / 1 pts

The fork() system call creates a new empty process.

- True

Correct!

- False

Question 5

0 / 1 pts

The Kernel-stub in a system call performs argument copy and validation before executing the requested operation.

orrect Answer

- True

ou Answered

- False

Question 6

1 / 1 pts

After handling a fault, program control resumes on the same instruction that caused the fault, and not the instruction after it.

Correct!

- True

False

Question 7

1 / 1 pts

Locks that utilize atomic test&set() instruction typically require busy-wait cycles.

Correct!

True

False

Question 8

5 / 5 pts

Match each term in the left column to the definition and /or description in the right column that fits best.

Correct! **medium term scheduler** — arbitrates on process su

Correct! **multiprogramming** — creates the illusion that

Correct! **asynchronous** — timing is not controlled b

Correct! **Shortest Job First (SJF) scheduling policy** — could result in starvation

Correct! **Round Robin scheduling policy** — CPU allocation with time

Question 9

5 / 5 pts

Match each term in the left column to the definition and /or description in the right column that fits best.

Correct!	**process control block**	contains information abc
Correct!	**polling**	technique to discover ha
Correct!	**context switching**	action performed by the
Correct!	**time sharing**	multiple processes (jobs
Correct!	**ready queue**	contains processes that

Question 10

1 / 1 pts

Process aging is:

- computing the next CPU burst time via a weighted exponential average of previous bursts.
- Correct! boosting a process' priority temporarily to get it scheduled to run.
- giving a process a longer quantum as it gets older.
- the measurement of elapsed CPU time during a process' execution.

Question 11

1 / 1 pts

Which scheduler gives each process an equal share of the CPU?

- Shortest Job First
- Round robin. (Correct!)
- Fixed Priority
- First Come First Served

Question 12

1 / 1 pts

Which scheduler relies on predicting the next process to schedule from among the set of ready processes based on their past performance?

- First Come First Served
- Round Robin
- Fixed Priority Queue
- Shortest Job First (Correct!)

Question 13

1 / 1 pts

A quantum is:

- the absolute minimum time that a process must run.

Correct!

- (•) the absolute maximum time that a process can run before being preempted.
- the amount of time that a process runs before it blocks on I/O.

Question 14

1 / 1 pts

What does a time-sharing system need to perform its function that a multiprogramming system does not?

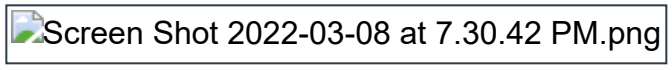
Correct!

- (•) Interval Timer
- System Call Mechanism
- Shorter Time Slices
- Kernel Mode Execution Privilege

Question 15

2.67 / 4 pts

Select **all probable outputs** produced by this code:



- [ ] x = 4

orrect Answer

- [ ] x = 2

Correct!

x = 3

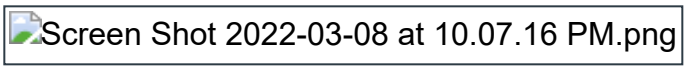
Correct!

x = 1

Question 16

0 / 4 pts

Review the program below and then the options presented below it:



D

orrect Answer

B

ou Answered

C

A

Explanation: After fork, the parent process most likely continues to execute the next line as normal while the child process takes a bit of time to get created and then execute. Thus, the first line is “This is CSCE 313.” The parent now waits for the child. The child prints the first “Howdy ”. execvp is executed next and prints “Howdy\n” to the console. After execvp, the child dies so the last print statement is not executed.

Question 17

1.67 / 5 pts

Indicate whether the listed output sequences for the program below are valid (Y) or invalid (N). Write Y for yes and N for no. Assume all system calls execute successfully.

```
int main()
{
        If (fork() == 0) {
                printf("a"); fflush(stdout);
        }
        else  {
        printf("b"); fflush(stdout);
        waitpid(-1, NULL, 0);
        }
        printf("c"); fflush(stdout);
        exit(0);
}
```

abcc = N

acbc = N

abc = Y

bacc = Y

cbca = N

bcac = Y

Answer 1:

ou Answered N

orrect Answer Y

orrect Answer yes

Answer 2:

ou Answered N

orrect Answer Y

orrect Answer yes

Answer 3:

ou Answered Y

orrect Answer N

orrect Answer no

Answer 4:

Correct! y

orrect Answer yes

Answer 5:

Correct! N

orrect Answer no

Answer 6:

ou Answered Y

orrect Answer N

orrect Answer no

Question 18

0 / 5 pts

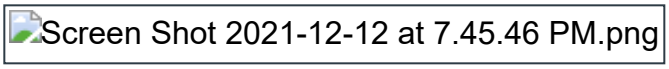
Your Answer:

In order to make sure that statement is printed to the terminal we would fork and then print dup2. Alternatively we can use fcntl().

Question 19

1 / 5 pts

Read the following code below followed by the description of a scenario:



Scenario: Assume the following sequence of events:

A. Main thread starts: it creates threads T1 and T2 and is then context switched by the scheduler (immediately after line 14 and before line 15).
B. Thread T2 is scheduled and runs until after it increments *count* (line 5) and is then context switched by the scheduler.
C. Thread T1 runs until it is context switched by the scheduler.
D. The scheduler resumes the main thread which runs until it get context switched.
E. Thread T2 gets scheduled and runs to completion.
F. The scheduler resumes the main thread which runs to completion (but doesn't exit yet).
G. Thread T1 runs to completion.

Answer the following questions:

1. **How is the above sequence of events possible? Write your reasoning in a few sentences.**
2. **Write the output of the program resulting from the above sequence.**

Your Answer:

1. The following sequence of events is possible due to multithreading. This allows events to run in parallel.

2.

Child : 0

Child : 1

Parent : 0

Parent : 1

Parent : 1

Question 20

1.7 / 5 pts

A computer system has three processes A, B, and C with the following characteristics:

- Process A arrives at time t=0ms, and uses the CPU for 100 ms before finishing.
- Process B arrives at time t=10ms. Process B loops nine times; for each iteration of the loop, B uses the CPU for 4 ms and then it does I/O for 6 ms.
- Process C arrives at time t=20ms and uses the CPU for 100 ms before finishing.

Identify the time when A, B and C will finish. Explain your answer with proper reasoning.

Assume the following:

1. **CPU scheduling scheme is preemptive and uses First Come First Served (FCFS) policy.**
2. **There is no overhead to doing a context switch.**

Your Answer:

A: 100ms in, because only CPU is uses and it is the first process to arrive

B: 100 + 9(4+6) = 190 ms , we add the first 100 ms from the first process since this is the second process to arrive then we add the time it takes this process to run. since it loops 9 times for the 4ms CPU task and 6ms I/O task we get a completion time of 100 + 9(4+6)

C: 100 + 9(4+6) + 100 = 290 ms , We first add the time the first process took 100 + 90 since this is the last process to arrive then we add the amount of time this process took to it 90.

Quiz Score: **28.04** out of 50