Quiz1

Due No due date **Points** 1.6 **Questions** 10
Available Sep 16 at 2pm - Sep 25 at 11:30pm 9 days **Time Limit** 20 Minutes

Instructions

Chapters 1-3

This quiz was locked Sep 25 at 11:30pm.

Attempt History

	Attempt	Time	Score
LATEST	Attempt 1	10 minutes	1 out of 1.6

Score for this quiz: **1** out of 1.6
Submitted Sep 16 at 2:14pm
This attempt took 10 minutes.

Question 1

0 / 0.1 pts

All of the following are possible approaches of inter-process communication except

orrect Answer ○ Multi-threading

○ File read/write

ou Answered ● Network messaging

○ pipe

Question 2

0 / 0.1 pts

Which of the following is FALSE regarding the operation: "file open, i.e., check if it exists, create if it does not exist, and then open the file"?

orrect Answer

○ It consists of multiple separate instructions

ou Answered

● It cannot be interrupted

○ It's defined in Unix system

○ It's designed to avoid potential errors

Question 3

0.1 / 0.1 pts

Which of the following functions is NOT a UNIX system call?

Correct!

● CreateProcess

○ exec()

○ wait()

○ signal()

Question 4

0.1 / 0.1 pts

Almost 90% system crashes were due to device drivers, not operating systems itself.

Correct!

● True

○ False

Question 5

0.07 / 0.2 pts

The three roles an operating system plays include referee , glue , and illusionist .

Answer 1:

Correct! Referee

Answer 2:

ou Answered glue

orrect Answer Illusionist

Answer 3:

ou Answered illusionist

orrect Answer Glue

Question 6

0.1 / 0.1 pts

A batch operating system works on a queue of tasks. It runs a simple loop: load, run, and unload each job in turn.

Answer 1:

Correct! batch

Question 7

0 / 0.2 pts

Most operating systems allocate both a program stack and interrupt stack for interrupted processes.

Answer 1:

ou Answered program

orrect Answer user

orrect Answer user level

orrect Answer user-level

Answer 2:

ou Answered interrupt

orrect Answer kernel

orrect Answer kernel level

orrect Answer kernel-level

Question 8

0.3 / 0.3 pts

Please fill in the following blanks in the code:

int child_pid = fork() ;

if(child_pid == 0) {

cout << "I am the child process " << getpid() << endl;

return 0;

}

```
else {
      cout << "I am the parent of child process " << child_pid <<
endl;
     return 0;
}
```

Answer 1:

Correct! fork()

Answer 2:

Correct! 0

Answer 3:

Correct! getpid()

Answer 4:

Correct! child_pid

Question 9

0.23 / 0.3 pts

We use mode switch to refer to any synchronous transfer of control from user mode to kernel. Three typical reasons for user to kernel mode transfer include interrupts , exceptions , system calls .

Answer 1:

ou Answered mode switch

orrect Answer trap

Answer 2:

Correct! interrupts

Answer 3:

ou Answered exceptions

orrect Answer processor exceptions

Answer 4:

Correct! system calls

Question 10

0.1 / 0.1 pts

An alternative to interrupts is polling : the kernel loops, checking each input/output device to see if an event has occurred that requires handling.

Answer 1:

ou Answered polling

orrect Answer pooling

Quiz Score: **1** out of 1.6