Pre-Lecture 12

Due Oct 2 at 9am	Points 8	Questions 6	Available until Oct 2 at 9am
Time Limit None	Allowed Atte	mpts 2	

Instructions

Take this quiz after you have watched the required videos and/or read the associated sections of the textbook. See <u>Lecture 12: Processes II</u>.

You may attempt this quiz twice. Incorrect responses are marked after each attempt. Correct answers are revealed at the start of class for this lecture.

Carefully note the deadline for responses. Submissions are not accepted after the deadline, and there is no grace period.

This quiz was locked Oct 2 at 9am.

Attempt History

	Attempt	Time	Score
LATEST	Attempt 1	127 minutes	8 out of 8

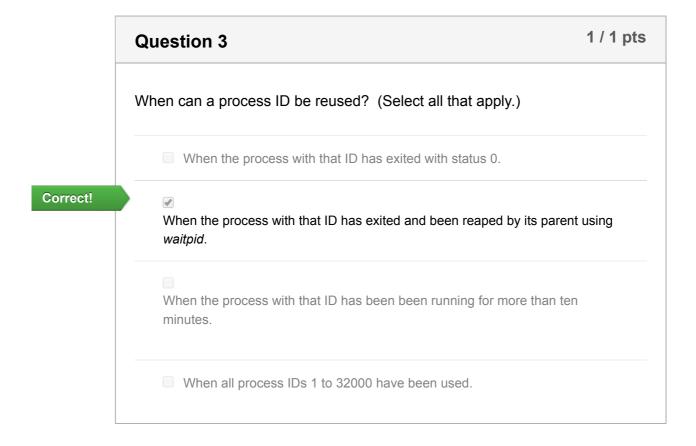
Score for this attempt: **8** out of 8 Submitted Oct 1 at 4:26pm

This attempt took 127 minutes.

```
1 / 1 pts
Question 1
What is printed by the following C program?
    #include <stdio.h>
    #include <stdlib.h>
    #include <unistd.h>
    #include <sys/types.h>
    #include <sys/wait.h>
    int main() {
      pid_t pid = fork();
      if(pid == 0)
        exit(42);
      else {
        int status = 0;
        waitpid(pid, &status, 0);
        printf("%d", WEXITSTATUS(status));
      }
    }
```

2019/11/16

Question 2 When a parent process that has forked a long-running child process exits, it causes the child process to end prematurely. True False



Question 4 1 / 1 pts What is the ID of the *init* process?

Correct! 1

orrect Answers 1

```
3 / 3 pts
Question 5
Consider the following C program:
    #include "csapp.h"
    /* Wait() is the same as Waitpid() with the
       pid and options parameters set to defaults.
       That is, it blocks until any child terminates. */
    int main() {
      if(Fork() == 0) {
        if(Fork() == 0)
          printf("a");
        else {
          pid_t pid;
          int status;
          if((pid = Wait(&status)) > 0)
            printf("b");
        }
      }
      else {
        printf("c");
        exit(0);
      printf("d");
      return 0;
Which of these outputs is possible? (Select all that apply.)
    abddc

✓ acdbd

    adbdc
    bdadc
    cadbd
```

Correct!

Correct!

Correct!

Question 6 1 / 1 pts

Consider the following C program *parent.c* (compiled to executable *parent*):

```
#include <unistd.h>

extern char** environ;

int main(int argc, char** argv) {
   argv[0] = "child";
   execve("child", argv, environ);
   return 0;
}
```

Also, consider the following C program *child.c* (compiled to executable *child*):

```
#include <stdio.h>
#include <unistd.h>

int main(int argc, char** argv) {
   int i;
   for(i = 0; i < argc; i++)
      printf("%s ", argv[i]);
   printf("%d", getpid());
   return 0;
}</pre>
```

Suppose that we run the parent program with this command:

```
./parent bird cat dog
```

Also, suppose that we use *ps* while the program is running to determine that the PID of *parent* in execution is 24712.

Exactly what is printed?

Correct!

child bird cat dog 24712

orrect Answers

child bird cat dog 24712

Quiz Score: 8 out of 8