## CSSE 332 -- OPERATING SYSTEMS

## Processes II

1\alliC
---------

**Question 1**. (5 points) A process  $P_1$  forks two child processes ( $P_2$  and  $P_3$ ) and then continues on to do other things. After some time,  $P_2$  completes, a couple of seconds after,  $P_3$  completes. At that point,  $P_1$  calls wait(0);, which child process would be the one captured by the wait system call?

- A.  $P_2$
- B.  $P_3$
- C. Cannot tell, it can be either of those child processes.

Question 2. (5 points) The \_\_\_\_\_\_ system call can be used to wait for \_\_\_\_\_ child process while the \_\_\_\_\_ system call can be used to wait for a \_\_\_\_\_ child process.

**Question 3**. Consider the snippet of code below.

```
1 int status;
  int rc = fork();
3 if (rc == 0) {
    // some child code goes here...
    // done, leave
    if(success) {
      exit(0);
    } else {
10
      exit(5);
11
    }
12 }
13
14 // parent code here.
15 // let's check on the child.
16 wait (&status);
17 if (status == 0) {
printf("My child completed successfully!\n");
19 } else {
20
  printf("My child was not successful!\n");
```

Fri Mar 14 2024 Page 1 of 4

(5 points	s) In the bo	ox below, re	write the pa	arent's cond	itional stat	ement to fi	x the two
(5 points above.	s) In the bo	ox below, re	write the pa	arent's cond	itional stat	ement to fi	x the two
	s) In the bo	ox below, re	write the pa	arent's cond	itional stat	ement to fi	x the two
	s) In the bo	ox below, re	write the pa	arent's cond	itional stat	ement to fi	x the two
	s) In the bo	ox below, re	write the pa	arent's cond	itional stat	ement to fi	x the two
	s) In the bo	ox below, re	write the pa	arent's cond	itional stat	ement to fi	x the two
	s) In the bo	ox below, re	write the pa	arent's cond	itional stat	ement to fi	x the two
	s) In the bo	ox below, re	write the pa	arent's cond	itional stat	ement to fi	x the two
	s) In the bo	ox below, re	write the pa	arent's cond	itional stat	ement to fi	x the two
	s) In the bo	ox below, re	write the pa	arent's cond	itional stat	ement to fi	x the two
	s) In the bo	ox below, re	write the pa	arent's cond	itional stat	ement to fi	x the two
	s) In the bo	ox below, re	write the pa	arent's cond	itional stat	ement to fi	x the two

Question 4. (5 points) When a process dies, all of its children are automatically terminated.

A. True.

B. False.

Fri Mar 14 2024 Page 2 of 4

Question 5. (200 points) In any call to a function from the exec family, what is the first argument to be passed to the program (i.e., second argument to exec)?
Question 6. (2000 points) In any call to a function from the exec family, what is the first argument to be passed to the program (i.e., second argument to exec)?
Question 7. (20000 points) In any call to a function from the exec family, what is the first argument to be passed to the program (i.e., second argument to exec)?
Question 8. (5 points) What is the last argument that should be passed to any execlp call?
Question 9. (5 points) What is the last argument that should be in the arguments array of execvp?

Fri Mar 14 2024 Page 3 of 4

Question 10. (5 points) Below is sample snippet of code written by a CSSE332 student.

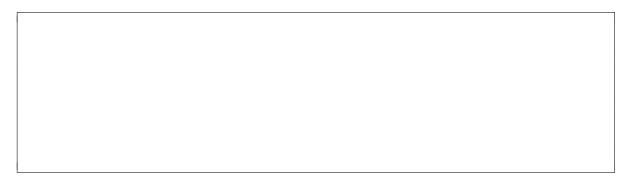
```
int rc = fork();
if(rc == 0) {
    // child process
    execlp("./buffalosay.bin", "./buffalosay.bin", arg_1, NULL);
printf("Done with buffaloysay.bin, let's do other stuff!\n");

// do something very important.

// done, now can leave.
exit(EXIT_SUCCESS);
}
```

The code snippet contains a significant bug. Identify the bug and suggest a way to fix it.

**Question 11**. (10 points) Please write down two **sentences** describing two new things that you learned in this session.



Question 12. (10 points) Please write down two things that you are still not very clear about, or any questions that you might have that the session did not go over or did not cover well.

Fri Mar 14 2024 Page 4 of 4