Exec and System Quizzes

Exercise

Indicate which of the following statements related to the system call exec are correct. Note that incorrect answers imply a penalty in the final score.

Choose one or more options:

1. ☐ In the case of correct execution of an exec, the instruction that follows the exec can be used to print a message of correct execution.
2. ☐ The execl used in the following way is correct: execl("cp","cp","./file1","./file2",(char*)0);
3. ☐ The execv has exactly two arguments, while the execve has exactly three arguments.
4. ☐ The execlp used in the following way is correct: execlp("cp","abcdefgh","./file1","./file2",(char*)0);
5. ☐ After an exec, the process identifier (pid) of the process is not changed.

Correct answer:
3, 4, 5

Exercise

1) Explain the effect of running the system call exec from a UNIX program.
2) Explain what differentiate the "l", "v", "p" and "e" versions.
3) Make an example of how the "v" and the "l" version can be used.
4) Why the system call exec should not return?

Answer:
1) The exec system call replaces the program associated with a process (code, data, etc.) with another one specified as a parameter, without replacing the PID (Process IDentifier) of the initial process.
2) It basically changes the prototype of the function and what it receives as input:
l: list -> the function receives a list of arguments.
v: vector -> the function receives a vector of arguments.
p: path -> the function receives only the name of the file (without the path). The file is searched by browsing the list of directories stored in the environment variable with name PATH.
e: environment -> the function receives a vector containing a list of environment variables.
3) Example execv:

```
char *cmd[] = {"ls", "-l", (char *)0};
execv("/bin/ls", cmd);
```

Example execl:

```
execl("/bin/ls", "ls", "-l", NULL);
```

4) The system call exec does not return to the caller program if executed with success. In the case of error, exec (in all its variants) returns -1.

Exercise

Suppose that the following program is executed passing the value 4 in the command line. Report the exact output generated by the program. Please, report the response in one line,, indicating all the messages and the values in output, which must be separated with only one space. Do not insert any other character in the response.

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
int main (int argc, char ** argv){
      char str[100];
      int i = atoi(argv[1]);
      setbuf(stdout, 0);
```

```
	printf("%d", i);
	sleep(1);
	if(i>0){
		sprintf(str, "%s %d", argv[0], i-1);
		system(str);
	}
	printf("End");
}
```

Correct Answers:
4 3 2 1 0 End End End End End

Exercise

Suppose you are running the following program. Report the output it generates, remembering to respect exactly the output format produced.
Assume that the program is running on an operating system where a sleep of 1 second is long enough to complete all the other tasks currently running.
Please, write the answer on a single line, indicating the various output messages/values separated by a single space. Do not enter any other character in the response.

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

int main(){
	int x;
	x=0;
	while (x<2 && fork()){
		if (!fork())
			execlp ("echo", "x++", "x", NULL);
		x++;
		sleep (1);
		system("echo x+x");
	}
}
```

Correct Answer:
`x x+x x x+x`

Exercise

If the following program is run with a single parameter equal to the integer value 5, i.e.,
`./pgrm 5`
how many characters 'E' and how many characters ‘e’ will be displayed on standard output?

```
int main (int argc, char *argv[]) {
	char str[20];
	int n = atoi (argv[1]);
	setbuf (stdout, 0);
	if (n>0) {
		printf ("E");
		sprintf (str, "%d", n-1);
		execlp (argv[0], argv[0], str, NULL);
	}
	printf ("e");
	return 1;
```

```
}
```

Choose JUST ONE option:

1. ☐ 5 'E', 5 'e'.
2. ☐ 1 'E', 5 'e'.
3. ☐ 1 'E', 1 'e'.
4. ☐ 5 'E', 1 'e'.
5. ☐ 6 'E', 6 'e'.

Correct Answer:
4

Exercise

Analyze the following segment of code. Assume that in a few milliseconds all output operations are completed. Indicate which is the output generated by the program. Note that incorrect answers imply a penalty in the final score.

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
int main(){
      int i;
      for (i=0; i<4 && !fork(); i++){
            if (fork()) {
                  sleep (1);
                  system ("echo i+");
            } else
                  execlp ("echo", "system", "i++", NULL);
      }
}
```

Choose JUST ONE option:

1. ☐ i++ 0+ i++ 1+ i++ 2+ i++ 3+
2. ☐ 0++ 0+ 1++ 1+ 2++ 2+ 3++ 3+
3. ☐ 0++ i+ 1++ i+ 2++ i+ 3++ i+
4. ☐ i++ i+ i++ i+ i++ i+ i++ i+
5. ☐ i+ i++ i+ i++ i+ i++ i+ i++

Correct answer:
4

Exercise

Analyze the following segment of code. Assume that in a few milliseconds all output operations are completed. Indicate how many characters 'E' are displayed on standard output when the program is executed passing 3 as its first argument. Report a single integer value in your response.

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <sys/wait.h>
int main (int argc, char *argv[]) {
```

```
  char str [50];
  int i;
  setbuf (stdout, 0);
  i = atoi (argv[1]);
  if (i<0)
    exit (0);
  if (fork () > 0) {
    if (fork () > 0) {
      sprintf (str, "echo -n E");
      system (str);
    }
    sprintf (str, "%d", i-2);
    printf ("E");
    execlp (argv[0], argv[0], str, NULL);
  }
  exit (0);
}
```

Correct Answer:
9