CYBR 401
Homework 3
Shell Project – Part 2

I. Overview

In this project you will be creating an interactive shell for your OS. This project will be broken into three parts, each building on the previous work. At the end, you will be able to run limited OS commands from within your operating system construct.

II. Part 2 Description

As we continue creating a shell, we need to be able to parse an input before feeding that into lower OS layers. As we parse input in our shell, the first argument will be the *command* to be run. For instance, in a call: *ls -h -a -l* the first argument *ls* is the function that needs to be called. Everything after is an argument that will be handed to that function.

In our shell, the entire command will be read in from the user after the command line prompt is printed. For instance, we will print *SUPERBASH$* and allow a user to enter a command. See the *ls example above.* Due to the limited nature of this example, your input command will be limited to only 200 characters and should have no additional spaces in the command.

Your shell should take commands and execute them just like any other command prompt interface. A command and arguments are entered, the command is executed, output is printed, and the shell resets for a new command. You must make your program exit of the exit command is entered from the command line.

You must use the fork(), execv(), and wait() functions to execute the commands. You may use the strcat(), strlen(), and strncmp() if you wish. All other functionality should be done by hand.

A starter file is included on Canvas that includes the recommended order in which to carry out the required tasks.

III. Requirements

- You must use the fork(), execv(), and wait() functions to execute OS calls.
- You must use your parsing function from part 1
- You may use strcat(), strlen(), and strncmp() if you would like

IV Rubric

Criteria	Pts
Uses fork(), execv(), and wait() to execute calls	/ 25
Uses part 1 parse function	/ 10
Correct Logic AND no compiler errors	/ 25
Code is organized, commented, and readable	/ 15
Output matches example below	/ 25

V. Helper Code Bit

Use this code to help determine if an input file exists.

```c
//See if a file exists
if(access(commandPath, F_OK ) != -1 ) {
    found = 1;
    printf("FOUND!\nString = '%s'\n", input);

    //Count pointers
    size = arraySize(parsedInput);
    printf("Number of pointers = %d\n", size);

    //Print addresses
    pointerPrint(parsedInput);
} else {
    commandPath = NULL;
}
```

VI. Example

```
aspanier@comeback-kid:~/Desktop/C Grading$ ./a.out
Address 0x7ffe718efb78
Base[0] Address 0x5558016b54a0 Pointer Value 0x5558016b52c0 String=PATH=/home/aspanier/.nvm/ve
Base[1] Address 0x5558016b5510 Pointer Value 0x5558016b52c8 String=/home/aspanier/.local/bin
Base[2] Address 0x5558016b5560 Pointer Value 0x5558016b52d0 String=/usr/local/sbin
Base[3] Address 0x5558016b55a0 Pointer Value 0x5558016b52d8 String=/usr/local/bin
Base[4] Address 0x5558016b55e0 Pointer Value 0x5558016b52e0 String=/usr/sbin
Base[5] Address 0x5558016b5620 Pointer Value 0x5558016b52e8 String=/usr/bin
Base[6] Address 0x5558016b5660 Pointer Value 0x5558016b52f0 String=/sbin
Base[7] Address 0x5558016b56a0 Pointer Value 0x5558016b52f8 String=/bin
Base[8] Address 0x5558016b56e0 Pointer Value 0x5558016b5300 String=/usr/games
Base[9] Address 0x5558016b5720 Pointer Value 0x5558016b5308 String=/usr/local/games
Base[10] Address 0x5558016b5760 Pointer Value 0x5558016b5310 String=/snap/bin

$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$

SUPER BASH $ls -h -a -l
Checking PATH=/home/aspanier/.nvm/versions/node/v18.12.1/bin/ls
Checking /home/aspanier/.local/bin/ls
Checking /usr/local/sbin/ls
Checking /usr/local/bin/ls
Checking /usr/sbin/ls
Checking /usr/bin/ls
FOUND!
String = 'ls -h -a -l'
Number of pointers = 5
Address 0x7ffe718efb18
Base[0] Address 0x5558016b6230 Pointer Value 0x5558016b6070 String=ls
Base[1] Address 0x5558016b6270 Pointer Value 0x5558016b6078 String=-h
Base[2] Address 0x5558016b62b0 Pointer Value 0x5558016b6080 String=-a
Base[3] Address 0x5558016b62f0 Pointer Value 0x5558016b6088 String=-l
total 48K
drwxrwxr-x  2 aspanier aspanier 4.0K Sep 29 15:26 .
drwxr-xr-x 51 aspanier aspanier 4.0K Sep 27 08:29 ..
-rwxrwxr-x  1 aspanier aspanier  17K Sep 29 15:26 a.out
-rw-rw-r--  1 aspanier aspanier 2.0K Sep 29 14:01 hw1.c
-rw-rw-r--  1 aspanier aspanier 1.2K Sep 27 08:55 lab_4.c
-rw-rw-r--  1 aspanier aspanier 6.2K Sep 29 15:26 test2.c
-rw-rw-r--  1 aspanier aspanier 2.5K Sep 29 15:04 test.c

$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$

SUPER BASH $
```