CYBR 401
Homework 4
Shell Project – Part 3

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 3 Description

The final part of the shell project will introduce the idea of piping into our shell functionality. There is a new starter code file for this project, however, we will start by bringing in all of the work you've done thus far. We will only be adding one new function in this part, but you will need to be sure it is integrated correctly. The best way to do this is to follow the HW4 starter file.

This part will implement all of the functions we've done this far, but we will introduce the pipe character. In this application, you will have a single pipe and then another command. There will be no extra space included. Due to the time consuming nature of input validation, *we are going to assume both commands are valid.*

III. Requirements

- 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.
- A pipe file is included to provide hints as to how to implement pipes.
- You must use the *dup2()* and *close() functions* to setup the pipes for the two children processes.

IV. Rubric

Criteria	Pts
Uses fork(), execv(), and wait() to execute calls	/ 5
Uses part 2 functionality	/ 5
Correct Logic AND no compiler errors	/ 5
Code is organized, commented, and readable	/ 5
Output matches example below	/ 5
The *exec2* function is correct	/ 35
The pipe is correctly implemented	/ 40

V. Example

```
aspanier@comeback-kid:~/Desktop/CYBR 401 - Fall 2023/Week 12$ ./a.out
Address 0x7fff8ae2b3a8
Base[0] Address 0x55e525ace4a0 Pointer Value 0x55e525ace2c0 String=PATH=/home/aspani
Base[1] Address 0x55e525ace510 Pointer Value 0x55e525ace2c8 String=/home/aspanier/.l
Base[2] Address 0x55e525ace560 Pointer Value 0x55e525ace2d0 String=/usr/local/sbin
Base[3] Address 0x55e525ace5a0 Pointer Value 0x55e525ace2d8 String=/usr/local/bin
Base[4] Address 0x55e525ace5e0 Pointer Value 0x55e525ace2e0 String=/usr/sbin
Base[5] Address 0x55e525ace620 Pointer Value 0x55e525ace2e8 String=/usr/bin
Base[6] Address 0x55e525ace660 Pointer Value 0x55e525ace2f0 String=/sbin
Base[7] Address 0x55e525ace6a0 Pointer Value 0x55e525ace2f8 String=/bin
Base[8] Address 0x55e525ace6e0 Pointer Value 0x55e525ace300 String=/usr/games
Base[9] Address 0x55e525ace720 Pointer Value 0x55e525ace308 String=/usr/local/games
Base[10] Address 0x55e525ace760 Pointer Value 0x55e525ace310 String=/snap/bin

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

SUPER BASH $ls|grep project
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'
Number of pointers = 2
Address 0x7fff8ae2b348
Base[0] Address 0x55e525acf450 Pointer Value 0x55e525acf290 String=ls
Checking PATH=/home/aspanier/.nvm/versions/node/v18.12.1/bin/grep
Checking /home/aspanier/.local/bin/grep
Checking /usr/local/sbin/grep
Checking /usr/local/bin/grep
Checking /usr/sbin/grep
Checking /usr/bin/grep
FOUND!
String = 'grep project'
Number of pointers = 3
Address 0x7fff8ae2b348
Base[0] Address 0x55e525acf630 Pointer Value 0x55e525acf470 String=grep
Base[1] Address 0x55e525acf670 Pointer Value 0x55e525acf478 String=project
starter_project3.c
```