Serge Strizhov ID: 922426905
Github: SergeStr CSC-415-03 Operating Systems

Assignment 3 – Simple Shell

Description:
This assignment is to write a C program that implements our own shell that runs on top of the regular command-line interpreter for Linux.

Approach:

Having little idea on how to utilize many of the required functions for this simple shell, I had to do a lot of research on how to utilize forks, pipes, dups and the many potential methods of accepting user input. For this purpose, I had utilized the linux manual page as well as studied its examples, as well as some lecture material.

Issues and Resolutions:

Firstly, I had to figure out how to have the simple shell accept user input, in which case I had to use fgets. I had thought of utilizing scanf, however, it was limited by the fact that there could be any amount of arguments, and scanf sees any strings delimited by spaces as their own ones, meaning that there is no way for me to alternate the scanf %s amount to manage for this. I could have used a complicated collection of flags for it, but I decided against it.

Secondly, I had to figure out how to separate the input by the arguments. To do this, I utilized strchr and strtok. strchr for initially counting the amount of spaces to find out the amount of arguments, and strtok to then separate all these arguments and fit them into an array whose size had been predetermined by the strchr test.

Thirdly, I had to set up a fork that would then execute the commands while the parent oversees it and outputs data on the child. I had a general of grasp on how to do it from the lectures, and after looking up manuals online as well as examples, I had figured out how to do it proper.

Fourthly, I had to figure out how to split up the original user input by the pipes and loop it in such a way that it would go over each function call separately. For this, I had to use strtok_r to select the currently being executed func, and then select the next one in the future.

Fifthly, I had to figure out how to pipe the data between the child and parent properly. At first, I was wholly confused at the concept, but after figuring out how the pipes sent data to each other, I made the child send output data to the parent that would then store it for the next function to utilize. To set up these output input lines, I had to use dup2s, which shifted the direction of the output into the pipe instead.

Analysis: No analysis for this assignment.

Screen shot of compilation:

```
student@student:~/Homework/csc415-assignment3-simpleshell-SergeStr$ make
gcc -c -o strizhov_serge_HW3_main.o strizhov_serge_HW3_main.c -g -I.
gcc -o strizhov_serge_HW3_main strizhov_serge_HW3_main.o -g -I. -l pthread
student@student:~/Homework/csc415-assignment3-simpleshell-SergeStr$
```

Screen shot(s) of the execution of the program:

```
student@student:~/Homework/csc415-assignment3-simpleshell-SergeStr$ make run
./strizhov_serge_HW3_main "Prompt> "
Prompt> ls --color
commands.txt  README.md                strizhov_serge_HW3_main.c
Makefile      strizhov_serge_HW3_main  strizhov_serge_HW3_main.o
Child 13622, exited with 0
Prompt> ls -l
total 60
-rw-rw-r-- 1 student student    66 Sep 24 18:59 commands.txt
-rw-rw-r-- 1 student student  1867 Sep 24 18:59 Makefile
-rw-rw-r-- 1 student student  7317 Sep 24 18:59 README.md
-rwxrwxr-x 1 student student 20816 Sep 26 15:29 strizhov_serge_HW3_main
-rw-rw-r-- 1 student student  5901 Sep 26 15:11 strizhov_serge_HW3_main.c
-rw-rw-r-- 1 student student 12000 Sep 26 15:29 strizhov_serge_HW3_main.o
Child 13626, exited with 0
Prompt> cat Makefile | wc -l -w
Child 13645, exited with 0
     64     304
Child 13646, exited with 0
Prompt> exit
student@student:~/Homework/csc415-assignment3-simpleshell-SergeStr$
```

more screenshots below

```
student@student:~/Homework/csc415-assignment3-simpleshell-SergeStr$ make run < commands.txt
./strizhov_serge_HW3_main "Prompt> "
commands.txt  README.md               strizhov_serge_HW3_main.c
Makefile      strizhov_serge_HW3_main  strizhov_serge_HW3_main.o
Prompt> Child 13651, exited with 0
"Hello World"
Prompt> Child 13652, exited with 0
total 72
drwxrwxr-x 3 student student  4096 Sep 26 15:29 .
drwxrwxr-x 5 student student  4096 Sep 24 18:59 ..
-rw-rw-r-- 1 student student    66 Sep 24 18:59 commands.txt
drwxrwxr-x 8 student student  4096 Sep 26 15:13 .git
-rw-rw-r-- 1 student student  1867 Sep 24 18:59 Makefile
-rw-rw-r-- 1 student student  7317 Sep 24 18:59 README.md
-rwxrwxr-x 1 student student 20816 Sep 26 15:29 strizhov_serge_HW3_main
-rw-rw-r-- 1 student student  5901 Sep 26 15:11 strizhov_serge_HW3_main.c
-rw-rw-r-- 1 student student 12000 Sep 26 15:29 strizhov_serge_HW3_main.o
Prompt> Child 13653, exited with 0
    PID TTY          TIME CMD
  12950 pts/0    00:00:00 bash
  13648 pts/0    00:00:00 make
  13649 pts/0    00:00:00 sh
  13650 pts/0    00:00:00 strizhov_serge_
  13654 pts/0    00:00:00 ps
Prompt> Child 13654, exited with 0
Prompt> Child 13655, exited with 0
     64     304
Child 13656, exited with 0
ls: cannot access 'foo': No such file or directory
Prompt> Child 13657, exited with 2
Prompt> Error: No input. Please enter a command.
student@student:~/Homework/csc415-assignment3-simpleshell-SergeStr$
```

```
student@student:~/Homework/csc415-assignment3-simpleshell-SergeStr$ make run
./strizhov_serge_HW3_main "Prompt> "
Prompt>
Error: No input. Please enter a command.
Prompt> student@student:~/Homework/csc415-assignment3-simpleshell-SergeStr$
```

(EOF test)