

Project 3: Unix Shell

The third and final project I made is the Unix shell. Most of the required functionalities are present however redirection, parallel commands and path-command haven't been completed. The wish.c consists of 4 functions: main(), Input(), parse-string() and execute().

- The main() -function handles execution checks; whether the program is run using command line arguments or a free-use mode where the user can type commands freely. It is also responsible for running the shell loops and starts other functions.
- Input() reads user input and saves it to a string. This is used when commands are read from user input rather than a file.
- Parse_String() parses command strings by removing white spaces and collecting commands in an array structure. Parse_String is used for reading both user input and files.
- Execute() is responsible for forking and executing given parsed commands using execvp() -system call. It also handles shell's own in-built commands: cd and exit.

The program can be given a file to read commands from rather than user input. The entire file is read and executed line by line.

There is no screenshot of the program since the code is too long, run examples are screenshotted below.

Example run with free use commands:

```
anton@anton-VirtualBox:~/Documents/Project_3$ gcc -o wish wish.c
anton@anton-VirtualBox:~/Documents/Project_3$ ./wish

wish> ls
tester.txt  wish  wish.c  wish.exe

wish> ls -l
total 52
-rw-rw-r-- 1 anton anton   80 heinä  28 17:48 tester.txt
-rwxrwxr-x 1 anton anton 17592 heinä  28 18:35 wish
-rw-rw-r-- 1 anton anton  5016 heinä  24 16:53 wish.c
-rwxrwxr-x 1 anton anton 17328 heinä  24 14:16 wish.exe

wish> cat tester.txt
ls -l
ls
mkdir test_directory
cd test_directory
touch test_file.txt
ls
exit nms

wish> mkdir test_directory

wish> ls
test_directory  tester.txt  wish  wish.c  wish.exe

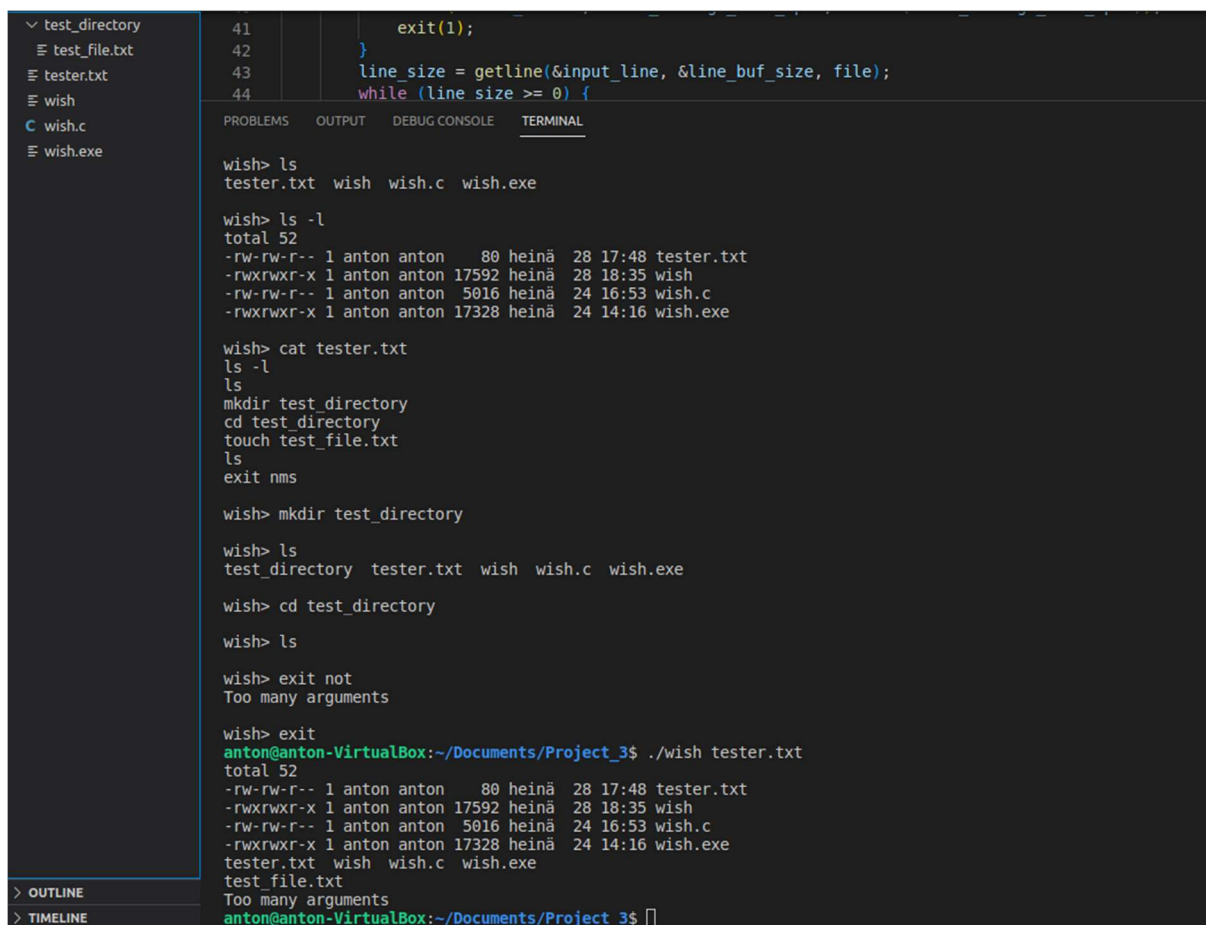
wish> cd test_directory

wish> ls

wish> exit not
Too many arguments

wish> exit
anton@anton-VirtualBox:~/Documents/Project_3$
```

Example run with a text file. (same tester.txt as in previous example)



```
41         exit(1);
42     }
43     line_size = getline(&input_line, &line_buf_size, file);
44     while (line_size >= 0) {

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

wish> ls
tester.txt wish wish.c wish.exe

wish> ls -l
total 52
-rw-rw-r-- 1 anton anton 80 heinä 28 17:48 tester.txt
-rwxrwxr-x 1 anton anton 17592 heinä 28 18:35 wish
-rw-rw-r-- 1 anton anton 5016 heinä 24 16:53 wish.c
-rwxrwxr-x 1 anton anton 17328 heinä 24 14:16 wish.exe

wish> cat tester.txt
ls -l
ls
mkdir test_directory
cd test_directory
touch test_file.txt
ls
exit nms

wish> mkdir test_directory

wish> ls
test_directory tester.txt wish wish.c wish.exe

wish> cd test_directory

wish> ls

wish> exit not
Too many arguments

wish> exit
anton@anton-VirtualBox:~/Documents/Project_3$ ./wish tester.txt
total 52
-rw-rw-r-- 1 anton anton 80 heinä 28 17:48 tester.txt
-rwxrwxr-x 1 anton anton 17592 heinä 28 18:35 wish
-rw-rw-r-- 1 anton anton 5016 heinä 24 16:53 wish.c
-rwxrwxr-x 1 anton anton 17328 heinä 24 14:16 wish.exe
tester.txt wish wish.c wish.exe
test_file.txt
Too many arguments
anton@anton-VirtualBox:~/Documents/Project_3$
```

The directory has updated and program exited after all commands had been processed.