



DẠI HỌC KHOA HỌC TỰ NHIÊN, ĐHQG-HCM
KHOA CÔNG NGHỆ THÔNG TIN
BỘ MÔN KHOA HỌC MÁY TÍNH

OPERATING SYSTEMS - HỆ ĐIỀU HÀNH

PROJECT REPORT 01

A SIMPLE SHELL

Giảng viên lý thuyết
Trần Trung Dũng

Giảng viên thực hành
Lê Giang Thanh

Nhóm thực hiện

Lê Nhựt Nam 18120061 18120061@student.hcmus.edu.vn
Đoàn Đăng Khoa 18120185 18120185@student.hcmus.edu.vn

Lời cảm ơn

Trong quá trình thực hiện đồ án này, em đã nhận được rất nhiều sự giúp đỡ cũng như hỗ trợ từ các thầy cô Trường Đại học Khoa học Tự nhiên, ĐHQG-HCM và các bạn bè trong lớp Hệ Điều Hành - CQ2018 21. Em xin bày tỏ lòng cảm ơn chân thành đến mọi người vì đã giúp đỡ hướng dẫn, chỉ bảo rất tận tình.

Đặc biệt, em xin bày tỏ lòng biết ơn sâu sắc đến các thầy cô khoa Công nghệ Thông tin, cụ thể hơn là thầy Trần Trung Dũng và các thầy hướng dẫn đã giảng dạy rất kĩ lưỡng, cung cấp nhiều slides, tài nguyên học tập cần thiết, tạo điều kiện tốt nhất để bản thân em có thể hoàn thành được đồ án này.

Dai hoc Khoa học Tự nhiên, DHQG-HCM.

Nhóm thực hiện

Tháng 10 năm 2020,

Mục lục

Lời cảm ơn	i
1 Tổng quan về đồ án	1
1.1 Mục đích thiết kế của đồ án	1
1.2 Bảng bảng đánh giá	1
2 Shell là gì?	1
3 Chu trình sống của một chương trình Shell	1
4 Các lệnh system calls có dùng trong đồ án	2
4.1 Lệnh fork()	2
4.2 Lệnh exec(), wait(), exit()	2
4.3 Lệnh pipe()	3
5 Đọc input người dùng	3
5.1 Parse command	3
5.2 Parse redirect command	3
5.3 Parse pipe command	3
6 Thực thi lệnh trong một tiến trình con	4
7 Tính năng Lịch sử với "!!"	4
8 Chuyển hướng đầu vào và đầu ra	4
8.1 Chuyển hướng đầu vào	4
8.2 Chuyển hướng đầu ra	4
9 Giao tiếp qua Pipe	4
10 Kiểm thử	5
10.1 Thực thi lệnh cơ bản	6
10.2 Thực thi lệnh dưới nền	7
10.3 Chuyển hướng đầu vào/ đầu ra và chức năng lịch sử	8
10.4 Giao tiếp hai lệnh thông qua Pipe	9
11 Tài liệu tham khảo	10

1 Tổng quan về đồ án

1.1 Mục đích thiết kế của đồ án

Với đồ án này, chúng em sẽ cài đặt lại một trình thông dịch lệnh UNIX đơn giản sao chép các chức năng của shell đơn giản (sh), tức sẽ hỗ trợ người dùng thực hiện một số câu lệnh đơn giản.

Các mô hình lệnh sẽ được cài đặt trong chương trình này là: thực thi cơ bản, thực thi dưới nền (background), điều hướng IO (>, <, >>), lịch sử, giao tiếp thông qua pipe.

Hiểu và trình bày được cơ bản về lifetime của một chương trình shell, UNIX fork(), exec(), wait(), dup2() và pipe() system calls.

Chương trình thiết kế trên vòng lặp vô hạn, sử dụng dấu nhắc lệnh có định dạng "<current day> <space> <current time> <space> <username>:<current directory working> >", được thiết kế gồm 4 phần chính:

- Executing command in a child process.
- Creating a history feature.
- Redirecting input and output (">" and "<")
- Communication via a pipe.

1.2 Bảng bảng đánh giá

Trình bày	Lệnh	Người thực hiện	Điểm tự đánh giá
	Các lệnh đơn giản với các tiến trình con	Nhựt Nam, Đăng Khoa	30
	Lệnh đơn giản với	Đăng Khoa	5
	Chuyển hướng Input	Nhựt Nam	15
	Chuyển hướng Outnput	Đăng Khoa	15
	Thực thi hai lệnh với một pipe	Đăng Khoa	15
Code Quality, Comments, Report the result in file (screenshot)		Nhựt Nam	20
Tổng điểm			100

2 Shell là gì?

Shell là chương trình người dùng đặc biệt, cung cấp giao diện cho người dùng sử dụng các dịch vụ hệ điều hành. Shell chấp nhận các lệnh có thể đọc được từ người dùng và chuyển đổi chúng thành thứ mà kernel có thể hiểu được.

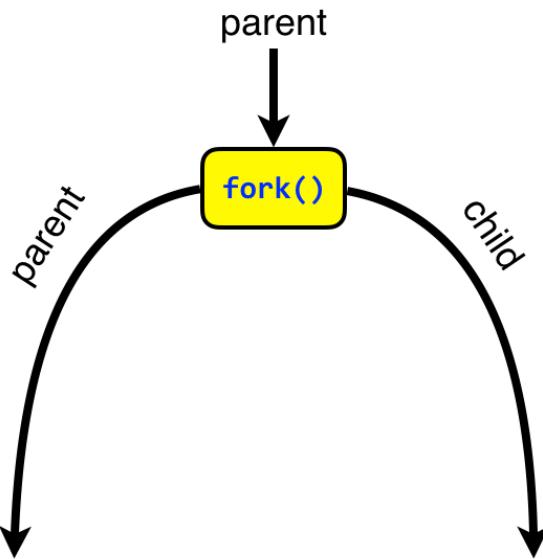
Theo linuxcommand.org shell được định nghĩa: " the shell is a program that takes commands from the keyboard and gives them to the operating system to perform"

3 Chu trình sống của một chương trình Shell

1. Khởi động chương trình shell.
2. Đọc input từ người dùng.
3. Paser lệnh được nhập từ người dùng. Nếu là lệnh "exit" hoặc "quit" (Tùy quy định thoát chương trình của người lập trình) thì thoát chương trình shell.
4. Thực thi lệnh và trả kết quả cho người dùng.
5. Quay lại bước 2.

4 Các lệnh system calls có dùng trong đồ án

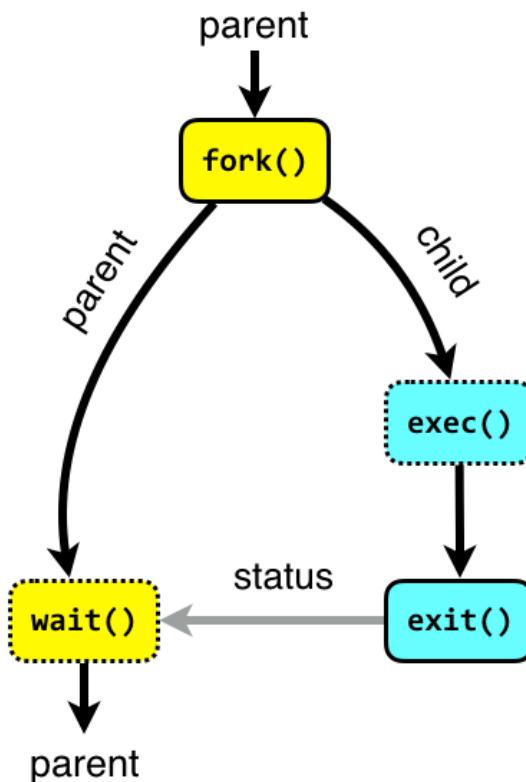
4.1 Lệnh fork()



Hình 1: Minh họa lệnh fork()

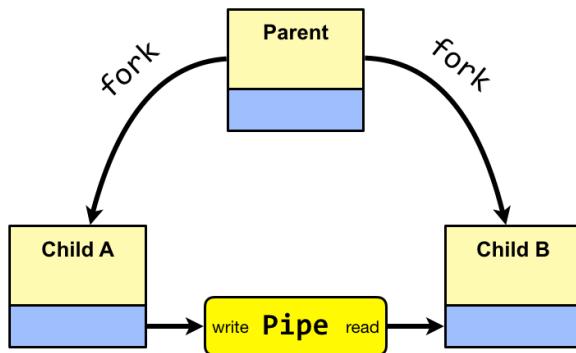
4.2 Lệnh exec(), wait(), exit()

execl - execv - execle - execve - execlp - execvp



Hình 2: Minh họa lệnh fork() - exec() - exit() - wait()

4.3 Lệnh pipe()



Hình 3: Minh họa pipe()

5 Đọc input người dùng

5.1 Parse command

```
void parse_command(char *input_string, char **argv, int *is_background)
```

5.2 Parse redirect command

Hàm kiểm tra

```
int is_redirect(char **argv)
```

Hàm parse

```
void parse_redirect(char **argv, char **redirect_argv, int redirect_index)
```

5.3 Parse pipe command

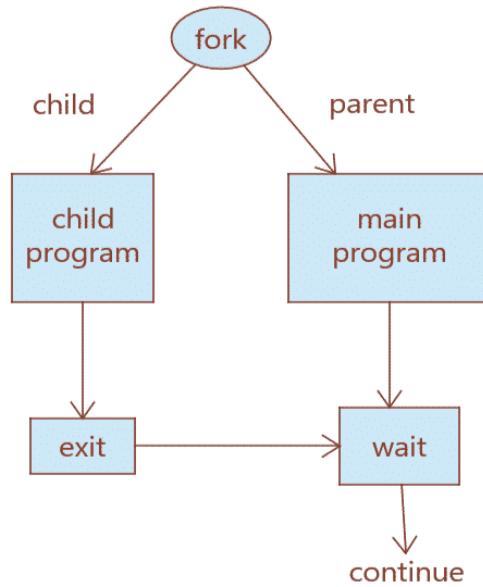
Hàm kiểm tra

```
int is_pipe(char **argv)
```

Hàm parse

```
void parse_pipe(char **argv, char **child01_argv,
char **child02_argv, int pipe_index)
```

6 Thực thi lệnh trong một tiến trình con



Hình 4: Thực thi lệnh trong một tiến trình con

7 Tính năng Lịch sử với "!!"

Lưu lại câu command trước đó vào một chuỗi, sau đó nếu người dùng nhập !! thì in ra chuỗi lịch sử đó và thực thi câu lệnh (Tương tự hoàn toàn với Linux Terminal)

8 Chuyển hướng đầu vào và đầu ra

8.1 Chuyển hướng đầu vào

```
void exec_child_overwrite_from_file(char **argv, char **dir)
```

8.2 Chuyển hướng đầu ra

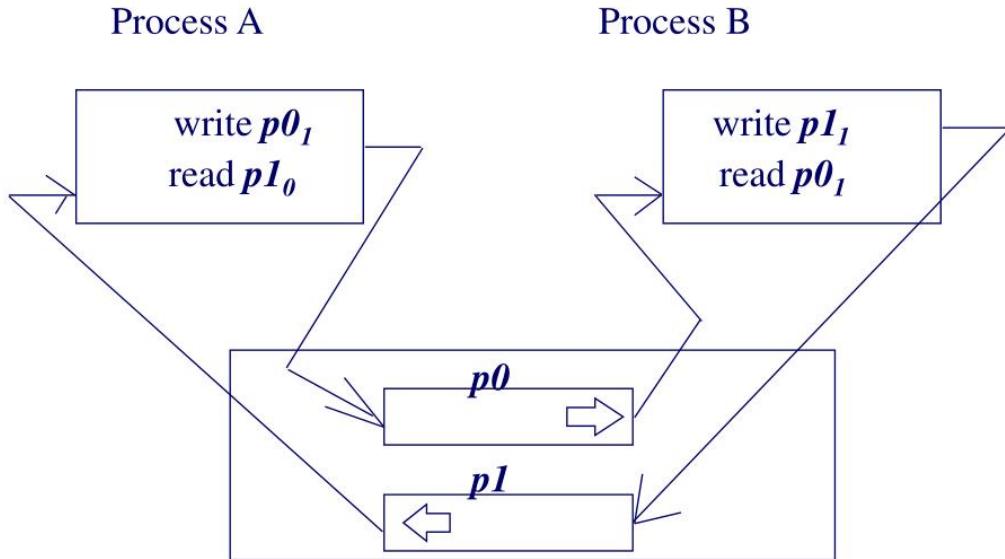
```
void exec_child_overwrite_to_file(char **argv, char **dir)
```

9 Giao tiếp qua Pipe

```
void exec_child_pipe(char **argv_in, char **argv_out)
```

Full Duplex Communication via Two Pipes

Two separate pipes, say $p0$ and $p1$



Hình 5: Minh họa cách giao tiếp thông qua Pipe

10 Kiểm thử

Yêu cầu: Linux Debian (có thể dùng Ubuntu, Zorin, Xubuntu, Kubuntu, ...), Make (4.2.1), GNU GCC Compiler $\geq 7.5.0$)

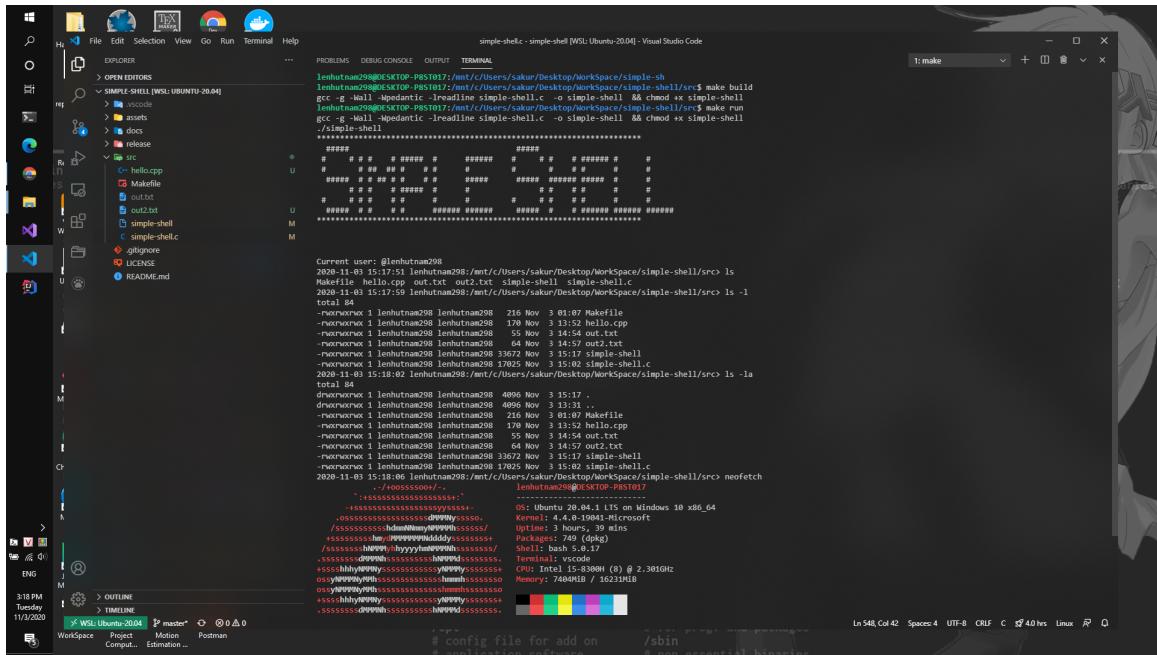
Lib: readline library "sudo apt-get install libreadline-dev"

Các build project:

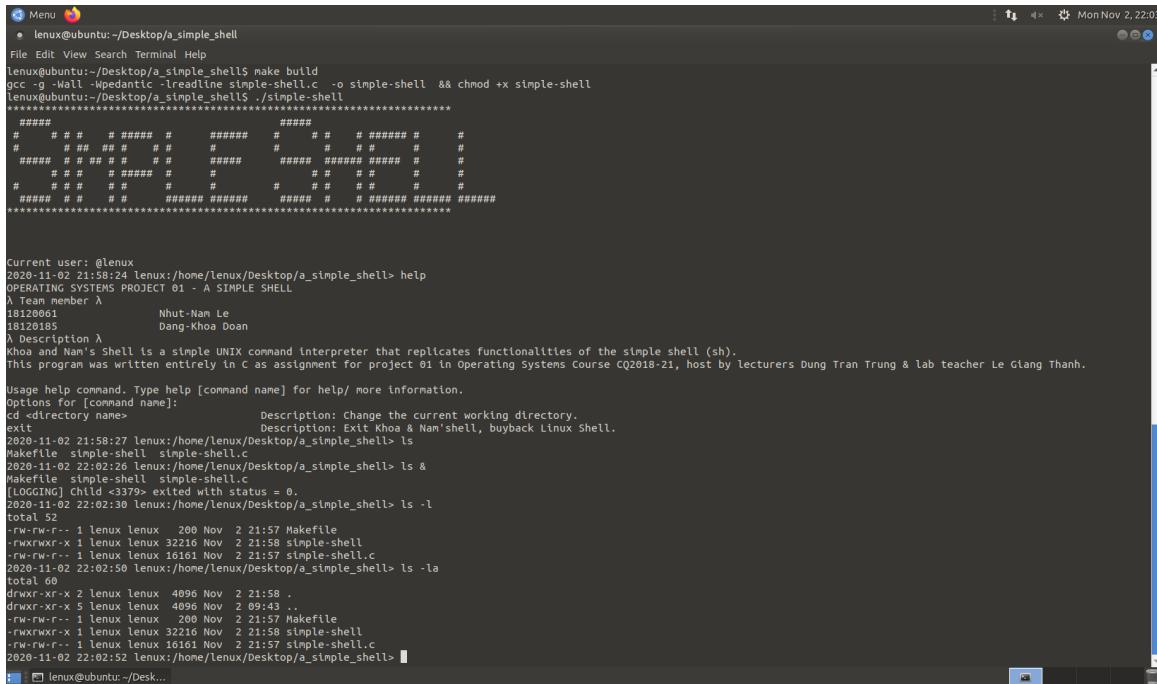
Bước 01: cd vào thư mục src của project

Bước 02: Mở terminal bằng cách nhấn tổ hợp phím Ctrl + Alt + T, gõ command "make build" để build project, sau đó gõ command "make run" để run chương trình simple shell

10.1 Thực thi lệnh cơ bản

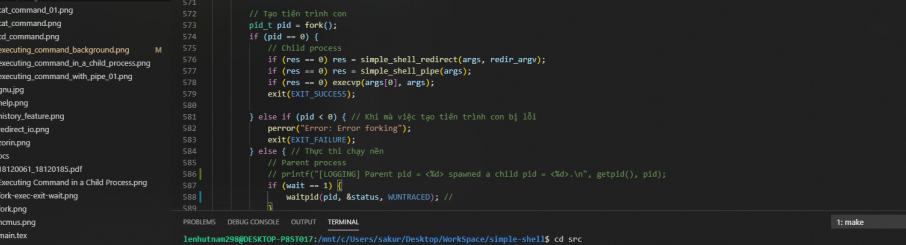


Hình 6: Thực thi lệnh đơn giản trên WSL - Ubuntu 20.04



Hình 7: Thực thi lệnh đơn giản trên Virtual Machine - Ubuntu 18.04

10.2 Thực thi lệnh dưới nền



The screenshot shows a Visual Studio Code interface with the following details:

- File Explorer:** Shows a folder named "SIMPLE-SHELL [WSL: UBUNTU-20.04]" containing files like "simple-shell.c", "cat_command_01.png", "cd_command.png", etc.
- Code Editor:** Displays the content of "simple-shell.c". The code implements a simple shell that forks a child process to execute user commands. It includes error handling for file operations and command execution.
- Terminal:** Shows the terminal output of the "make run" command on an Ubuntu 20.04 WSL instance. The output includes the compilation of "simple-shell.c" and its execution, displaying a shell prompt with the user "sakur".
- Status Bar:** Shows the current user as "sakur" and the date/time as "2028-11-03 23:58:14".

Hình 8: Thực thi lệnh dưới nền trên WSL - Ubuntu 20.04

```
lenox@ubuntu: ~/Desktop/a_simple_shell
```

```
File Edit View Search Terminal Help
```

```
# # # # ##### # ##### # # # # ##### # #
# # # # # # # # # # ##### # ##### # # # #
##### # # # # # # # ##### ##### ##### # # #
# # # # # # # # # # ##### # # # # # #
# # # # # # # # # ##### # # ##### ##### #####
*****
```

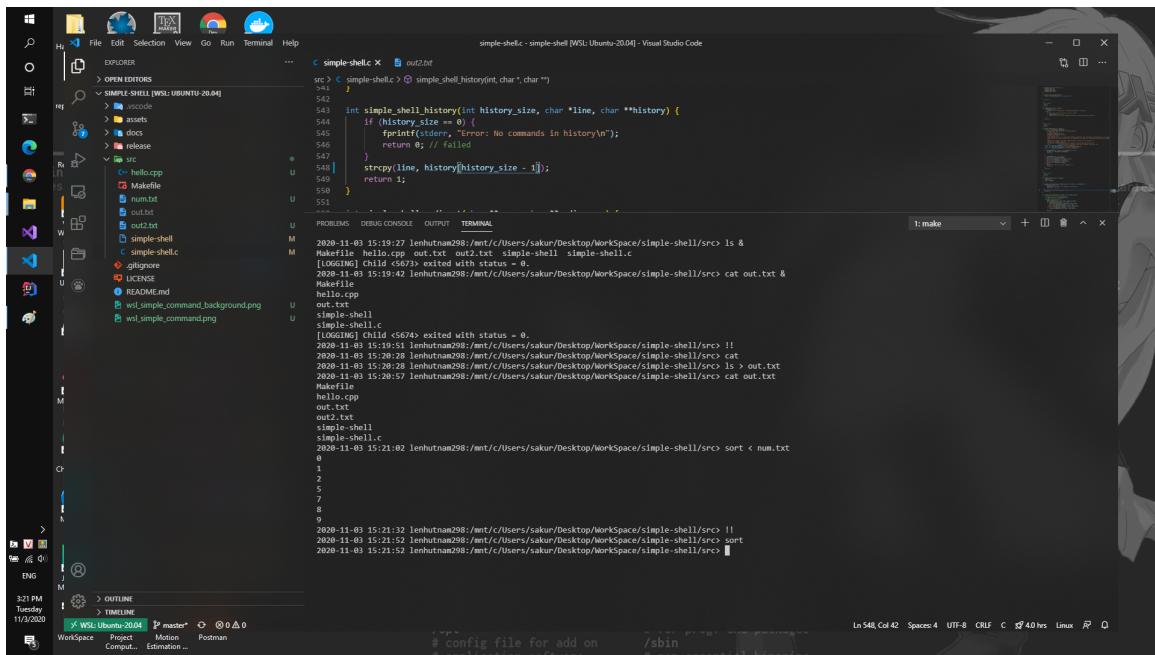
```
Current user: lenox
2020-11-03 09:02:38 lenox:/home/lenox/Desktop/a_simple_shell> ls &
2020-11-03 09:02:42 lenox:/home/lenox/Desktop/a_simple_shell> hello.c hellocpp.cpp list_in_dir.txt list_number.txt Makefile simple-shell simple-shell.c
cd ..
2020-11-03 09:02:45 lenox:/home/lenox/Desktop> ls &
2020-11-03 09:02:49 lenox:/home/lenox/Desktop> a_simple_shell demo_simple_shell simple-shell
ls -l
total 12
drwxr-xr-x 2 lenox lenox 4096 Nov  3 09:02 a_simple_shell
drwxrwxr-x 2 lenox lenox 4096 Oct 23 08:50 demo_simple_shell
drwxrwxr-x 4 lenox lenox 4096 Nov  2 09:46 simple-shell
2020-11-03 09:02:51 lenox:/home/lenox/Desktop> cd a_simple_shell
2020-11-03 09:03:01 lenox:/home/lenox/Desktop/a_simple_shell> ls
hello, hellocpp.cpp list_in_dir.txt list_number.txt Makefile simple-shell simple-shell.c
2020-11-03 09:03:03 lenox:/home/lenox/Desktop/a_simple_shell> cat hello.c &
2020-11-03 09:03:23 lenox:/home/lenox/Desktop/a_simple_shell> ls
hello, hellocpp.cpp list_in_dir.txt list_number.txt Makefile simple-shell simple-shell.c
2020-11-03 09:03:28 lenox:/home/lenox/Desktop/a_simple_shell> cat hello.c
2020-11-03 09:04:30 lenox:/home/lenox/Desktop/a_simple_shell> cat hellocpp.cpp &
2020-11-03 09:05:16 lenox:/home/lenox/Desktop/a_simple_shell> #include <iostream>
#include <functional>

int main(int argc, char* argv[]) {
    [out = std::ref(std::cout << "Hello ")] () { out.get() << "C++\n"; }();
    return 0;
}

ls -l
total 68
-rw-rw-r-- 1 lenox lenox   0 Nov  3 00:46 hello.c
-rw-rw-r-- 1 lenox lenox  171 Nov  2 22:05 hellocpp.cpp
-rwxr--r-- 1 lenox lenox  82 Nov  2 22:07 list_in_dir.txt
-rwxr--r-- 1 lenox lenox  23 Nov  2 22:07 list_number.txt
-rwxr--r-- 1 lenox lenox 280 Nov  3 09:02 simple-shell
-rwxrwxr-x 1 lenox lenox 31648 Nov  3 09:02 simple-shell.c
-rw-rw-r-- 1 lenox lenox 20190 Nov  3 09:02 simple-shell.c
2020-11-03 09:05:21 lenox:/home/lenox/Desktop/a_simple_shell>
```

Hình 9: Thực thi lệnh dưới nền trên Virtual Machine - Ubuntu 18.04

10.3 Chuyển hướng đầu vào/ đầu ra và chức năng lịch sử



Hình 10: Chuyển hướng đầu vào/ đầu ra và chức năng lịch sử trên WSL - Ubuntu 20.04

```
● lenux@ubuntu: ~/Desktop/a_simple_shell
File Edit View Search Terminal Help
-rw-rw-r-- 1 lenux lenux 200 Nov 2 21:57 Makefile
-rwxrwxr-x 1 lenux lenux 32216 Nov 2 21:58 simple-shell
-rw-rw-r-- 1 lenux lenux 10616 Nov 2 21:57 simple-shell.c
2020-11-02 22:02:52 lenux:/home/lenux/Desktop/a_simple_shell> ls
hello.cpp list_number.txt Makefile simple-shell simple-shell.c
2020-11-02 22:05:21 lenux:/home/lenux/Desktop/a_simple_shell> cat list_number.txt
1
3
5
2
7
6
8
10
0
-5

2020-11-02 22:05:37 lenux:/home/lenux/Desktop/a_simple_shell> sort < list_number.txt
0
1
2
3
5
-5
6
7

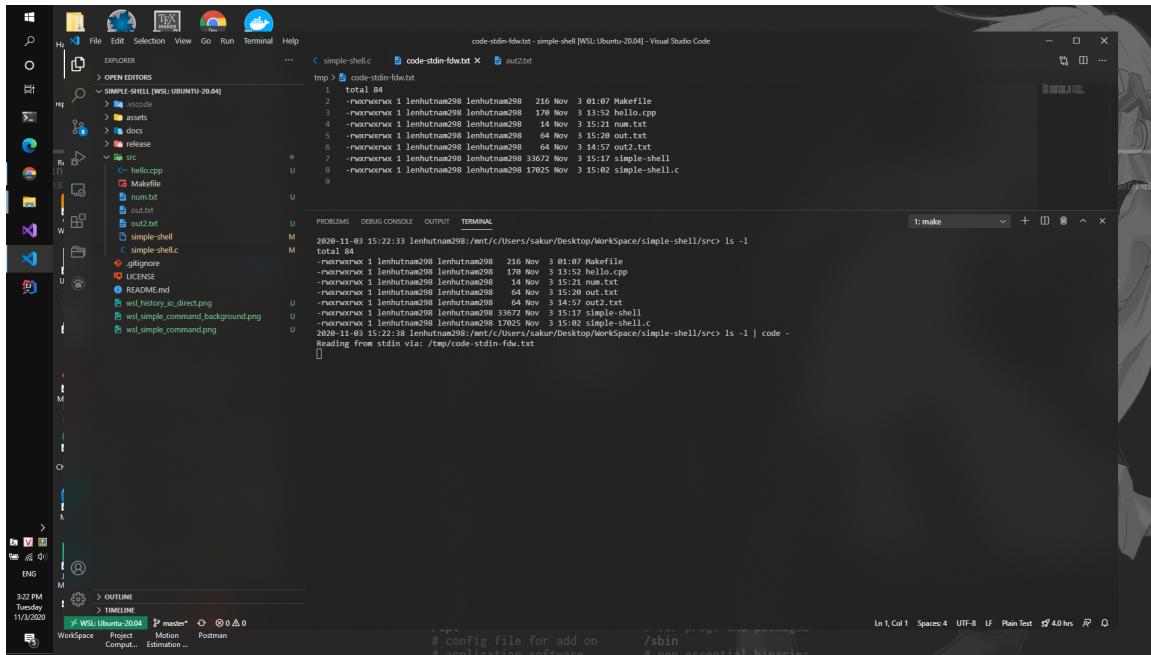
2020-11-02 22:06:05 lenux:/home/lenux/Desktop/a_simple_shell> cat hello.cpp
#include <iostream>
#include <functional>

int main(int argc, char* argv[]) {
    [out = std::ref(std::cout << "Hello ")] () { out.get() << "C++\n"; }()
    return 0;
}

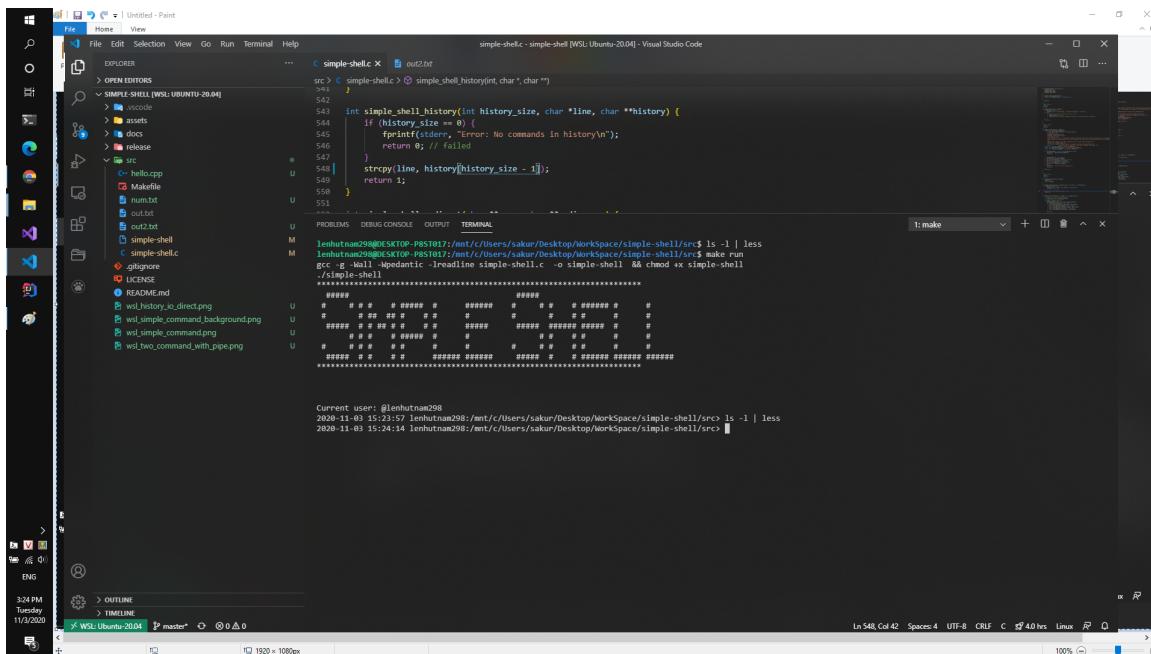
2020-11-02 22:06:39 lenux:/home/lenux/Desktop/a_simple_shell> ls > list_in_dir.txt
2020-11-02 22:07:01 lenux:/home/lenux/Desktop/a_simple_shell> cat list_in_dir.txt
hello.cpp
list_in_dir.txt
list_number.txt
Makefile
simple-shell
simple-shell.c
2020-11-02 22:07:10 lenux:/home/lenux/Desktop/a_simple_shell>
```

Hình 11: Chuyển hướng đầu vào/ đầu ra trên Virtual Machine - Ubuntu 18.04

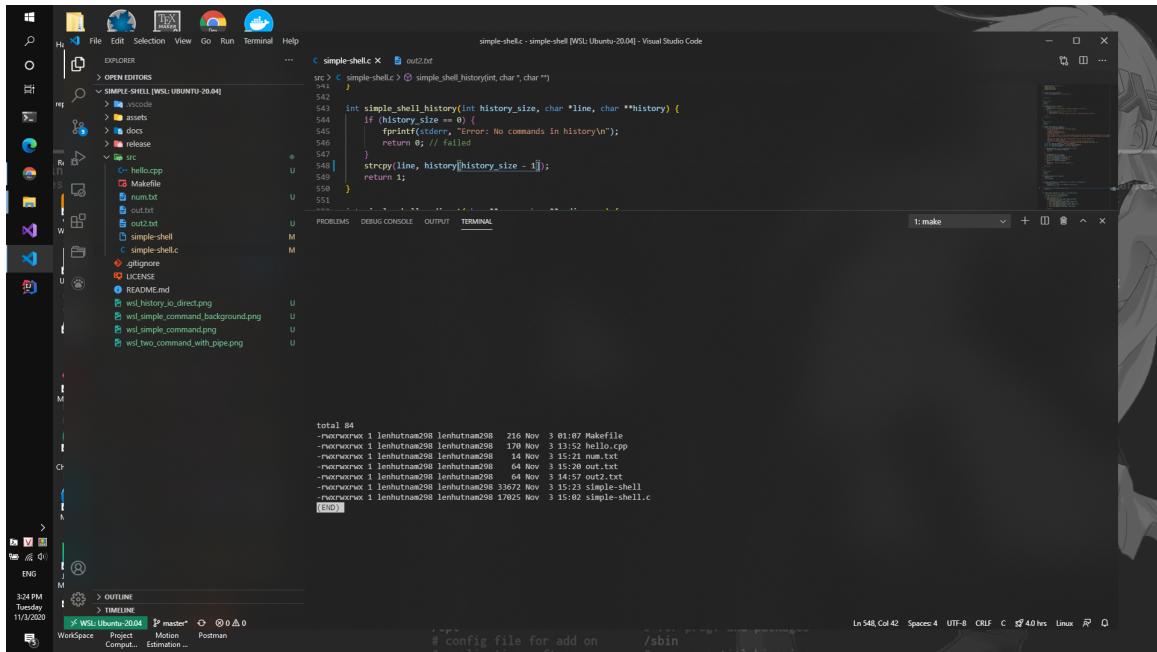
10.4 Giao tiếp hai lệnh thông qua Pipe



Hình 12: Test: ls -l | code-



Hình 13: Test: ls -l | less



Hình 14: Test: ls -l | less

Hình 15: Test thực thi Pipe trên VM Ubuntu 18.04

11 Tài liệu tham khảo

- [1] Linux man pages - System calls
[2] EnthusiastiCon - Stefanie Schirmer "OMG building a shell in 10 minutes"