HO CHI MINH CITY UNIVERSITY OF TECHNOLOGY

Faculty of Computer Science & Engineering

Operating System
Lab 1

Instructor: Mrs. Le Thanh Van

Students: *Cao Hoang Kiet - 2053165*

Ho Chi Minh City, February 2022

Contents

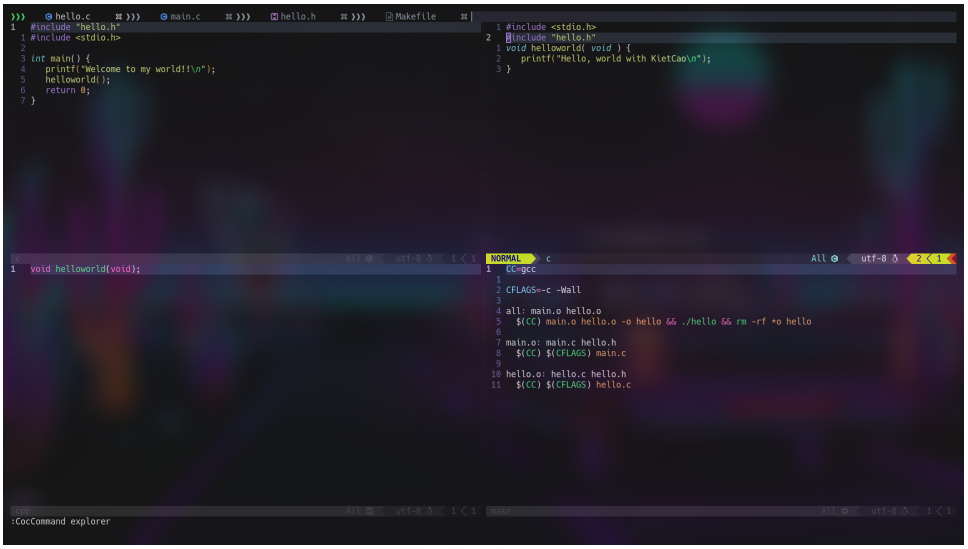
1 Questions

1.1 What are the advantages of Makefile? Give examples?

1.1.1 Advantages

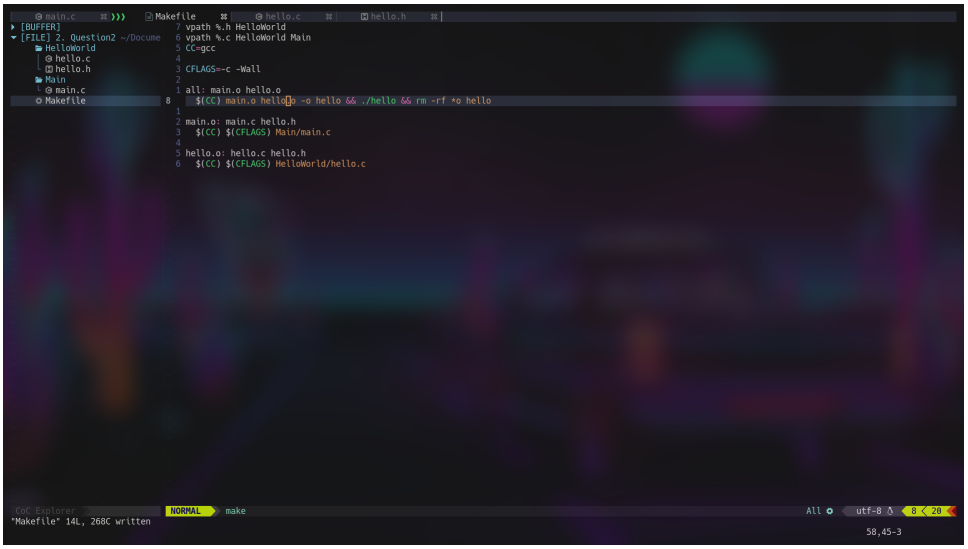
- It makes codes more concise and clear to read and debug.
- No need to compile entire program every time whenever you make a change to a functionality or a class. Makefile will automatically compile only those files where change has occurred.
- Generally, in long codes or projects, Makefile is widely used in order to present project in more systematic and efficient way.

1.1.2 Examples

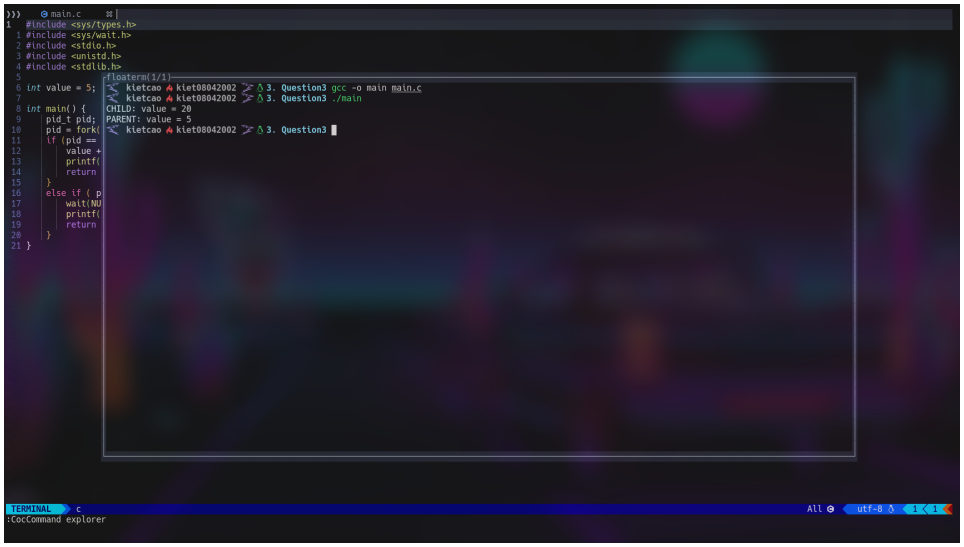


1.2 In case of source code files located in different places, how can we write a Makefile?

We use vpath to specify where files exist



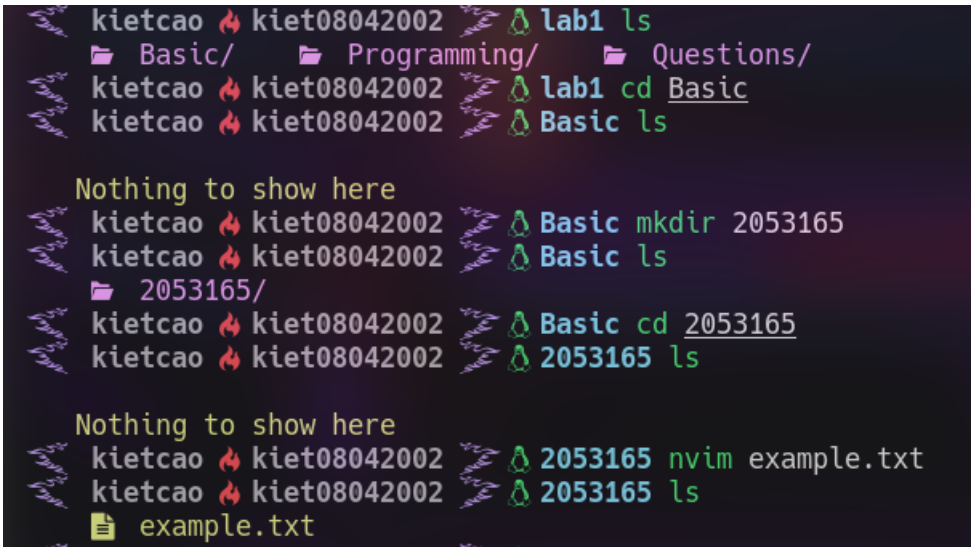
1.3 What the output will be at LINE A? Explain your answer.



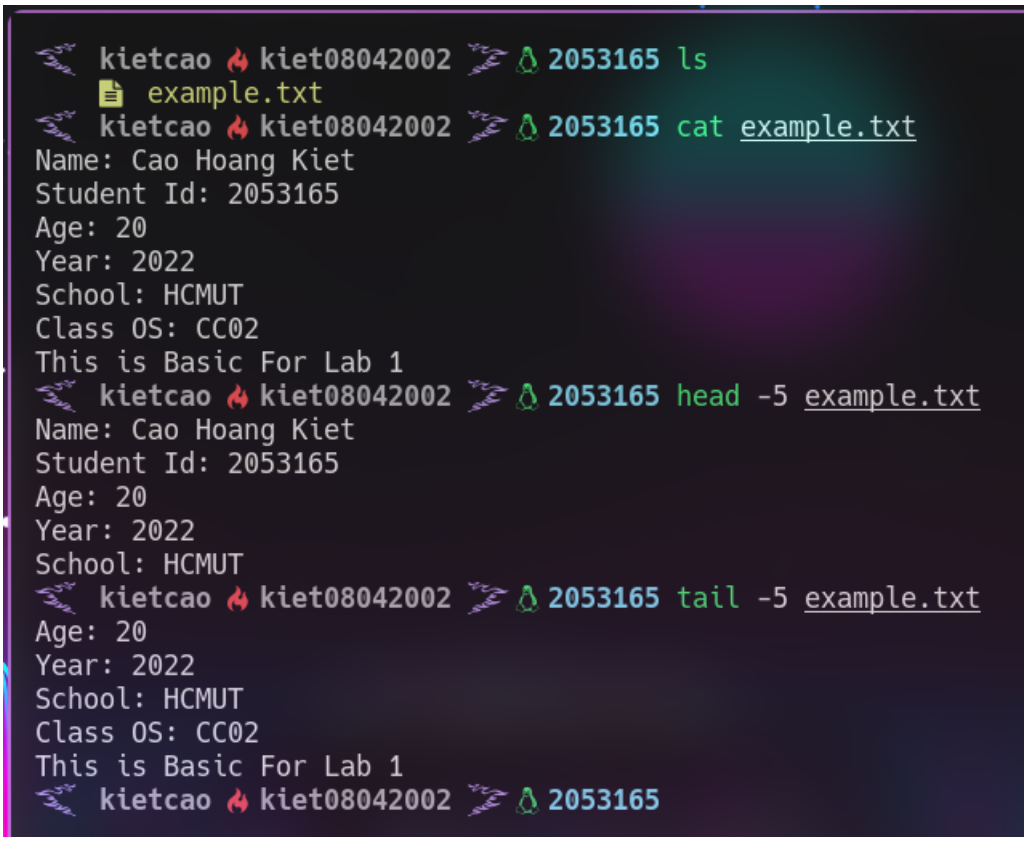
Answer: The output at LINE A is still 5 because the child just updates its copy of value. So when control returns to the parent, its value is keep at 5.

2 Basic Commands

2.1 Create a new directory entitled with your student ID + Create a new file example.txt under folder <StudentID>

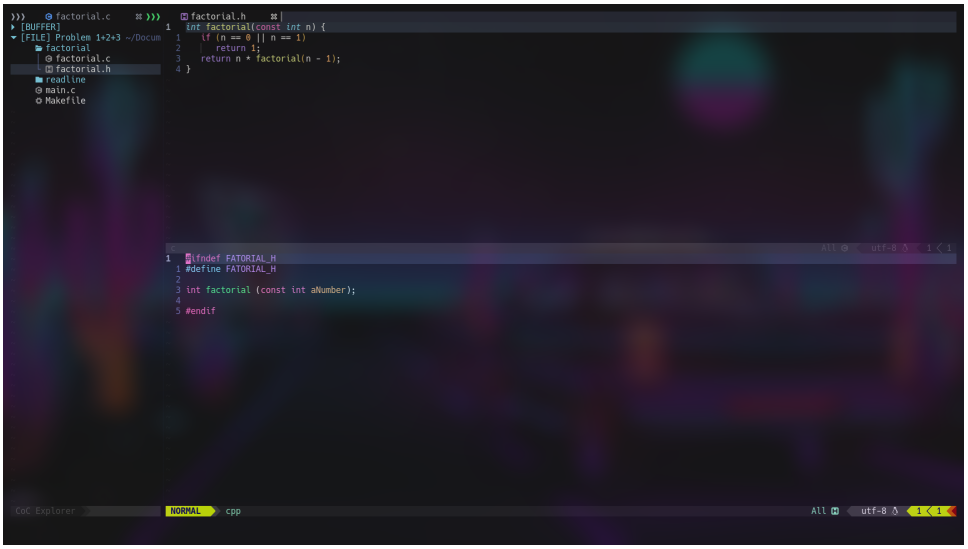


2.2 List the content of <student-ID> directory + View the content of file example.txt + Show first 5 lines from example.txt + Show last 5 lines from example.txt.



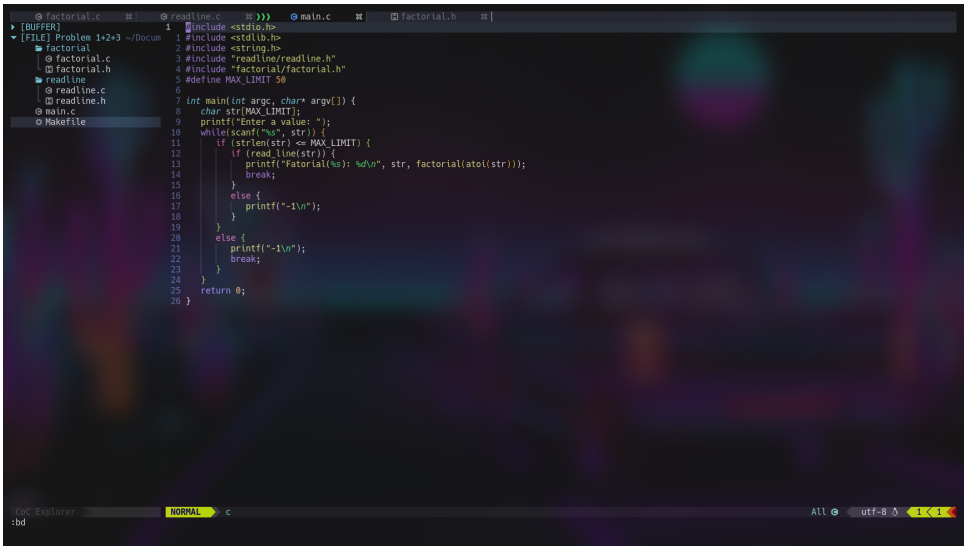
3 Programming

3.1 Problem 1



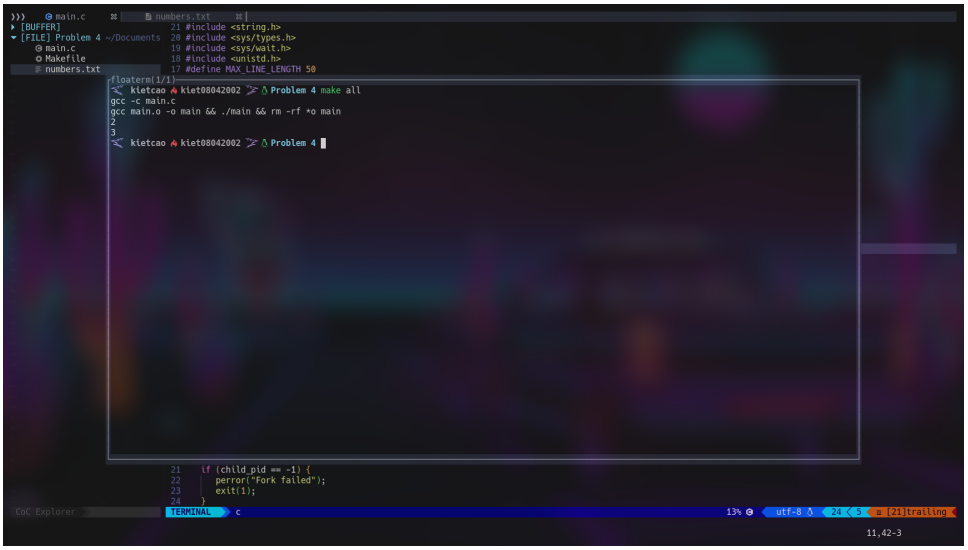
3.2 Problem 2

3.3 Problem 3



3.4 Problem 4


main.c
int main(int argc, char **argv)
{
char line[MAX_LINE_LENGTH] = {0};
unsigned int sizeArr = 0;
int arr[MAX_LINE_LENGTH], num1 = 0;
FILE *file = fopen("numbers.txt", "r");
if (!file)
{
perror("numbers.txt");
return EXIT_FAILURE;
}
while (fgets(line, MAX_LINE_LENGTH, file))
{
sizeArr++;
arr[sizeArr-1] = atoi(line);
}
if (fclose(file))
{
return EXIT_FAILURE;
perror("numbers.txt");
}
pid_t child_pid;
child_pid = fork();
if (child_pid == -1) {
perror("Fork failed");
exit(1);
}
if (child_pid == 0) {
countDivisibleNumbers(arr, sizeArr, &num1, 3);
num1 != 0 ? printf("%d\n", num1) : -1;
fflush(stdout);
}
else {
wait(NULL);
countDivisibleNumbers(arr, sizeArr, &num1, 2);
num1 != 0 ? printf("%d\n", num1) : -1;
fflush(stdout);
}
}
NORMAL c
Bot utf-8 25 5 [21]trailing


main.c
numbers.txt
[BUFFER]
[FILE] Problem 4 ~/Documents
main.c
Makefile
numbers.txt
#include <string.h>
#include <sys/types.h>
#include <sys/wait.h>
#include <unistd.h>
#define MAX_LINE_LENGTH 50
floaterm[1/1]
kietcao kiet08042002 Problem 4 make all
gcc -c main.c
gcc main.o -o main && ./main && rm -rf *o main
2
3
kietcao kiet08042002 Problem 4
if (child_pid == -1) {
perror("Fork failed");
exit(1);
}
CoC Explorer
TERMINAL c
13% utf-8 24 5 [21]trailing
11,42-3