

### (1) Task 1

(1) Write a C program that uses standard I/O libraries to display the contents of text files. The program is compiled and linked by the make tool, which requires the generation of the.o file first, and then the generation of the executable file, and the function of deleting the intermediate file (.o) in the makefile file.

```
deng@ubuntu: ~/桌面
deng@ubuntu:~/桌面$ vim text
deng@ubuntu:~/桌面$ vim c1.c
deng@ubuntu:~/桌面$ gcc -o hello1 c1.o
gcc: error: c1.o: 没有那个文件或目录
gcc: fatal error: no input files
compilation terminated.
deng@ubuntu:~/桌面$ gcc -c c1.c
deng@ubuntu:~/桌面$ gcc -o hello1 c1.o
deng@ubuntu:~/桌面$ ./hello1 text.txt
B22131014 dengqixiang
i love you all!
```

### (2) Task 2

(2) Write a C program that displays all the file names in the current directory. The program is compiled and linked by the make tool, which requires the generation of the.o file first, and then the generation of the executable file, and the function of deleting the intermediate file (.o) in the makefile file.

```
deng@ubuntu:~/桌面$ vim c2.c
deng@ubuntu:~/桌面$ gcc -c c2.c
deng@ubuntu:~/桌面$ gcc -o hello2 c2.o
deng@ubuntu:~/桌面$ ./hello2
error: failed to open directory
deng@ubuntu:~/桌面$ vim c2.c
deng@ubuntu:~/桌面$ ./hello2 .
.
hello2
c1.o
c1.c
.1.c.swp
.33.cpp.swp
text.txt
c2.c
..
hello1
c2.o
study
```

### (3) Task 3

(3) Write a C program that changes the working directory of the current process. The program is compiled and linked by the make tool, which requires the generation of the.o file first, and then the generation of the executable file, and the function of deleting the intermediate file (.o) in the makefile file.

```
deng@ubuntu:~/桌面$ vim c3.c
deng@ubuntu:~/桌面$ gcc -c c3.c
deng@ubuntu:~/桌面$ gcc -o hello3 c3.o
deng@ubuntu:~/桌面$ ./hello3
Current Directory: /home/deng/桌面
success: changed to /home
New Directory: /home
```