

(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.

```
wang@wang-virtual-machine:~/linux$ gcc -c b22040706.c
wang@wang-virtual-machine:~/linux$ gcc -o hello1 b22040706.o
wang@wang-virtual-machine:~/linux$ ./hello1 1.txt
B22040706
wang@wang-virtual-machine:~/linux$
```

(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.

```
wang@wang-virtual-machine:~/linux$ gcc -c b22040706.c
wang@wang-virtual-machine:~/linux$ gcc -o hello2 b22040706.o
wang@wang-virtual-machine:~/linux$ ./hello2 ~/linux
b22040706.o
hello1
demo
b22040706.c
.
test.c
..
hello2
1.txt
wang@wang-virtual-machine:~/linux$
```

(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.

```
wang@wang-virtual-machine:~/linux$ gcc -c b22040706.c
wang@wang-virtual-machine:~/linux$ gcc -o hello3 b22040706.o
wang@wang-virtual-machine:~/linux$ ./hello3
/home/wang/linux
success
/home
wang@wang-virtual-machine:~/linux$
```

