(1) Task 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.

```
  GNU nano 8.3                          c1.c
#include <stdio.h>
int main(int argc, char* argv[])
{
    char buf[1024] = { 0 };
    FILE* fp = fopen(argv[1],"r");
    if (argc < 2)
    {
        printf("please input source file!\n");
    }
    if (fp == NULL)
    {
        printf("open source %s failed\n", argv[1]);
        return -1;
    }
    while (fgets(buf,1024, fp))
    {
        printf("%s\n", buf);
    }
    return 0;
}
```

```
  GNU nano 8.3                          Makefile
hello1:c1.o
        gcc -o hello1 c1.o
c1.o:c1.c
        gcc -c c1.c
clean:
        rm -rf *.o
```

```
  GNU nano 8.3                          test.txt
This is the first line.
This is the second line.
```

```
B22040104谢淡恬@ubuntu64:~/exp_3/task1$ nano c1.c
B22040104谢淡恬@ubuntu64:~/exp_3/task1$ nano Makefile
B22040104谢淡恬@ubuntu64:~/exp_3/task1$ make
gcc -c c1.c
gcc -o hello1 c1.o
B22040104谢淡恬@ubuntu64:~/exp_3/task1$ nano test.txt
B22040104谢淡恬@ubuntu64:~/exp_3/task1$ ./hello1 test.txt
This is the first line.

This is the second line.

B22040104谢淡恬@ubuntu64:~/exp_3/task1$ make clean
rm -rf *.o
B22040104谢淡恬@ubuntu64:~/exp_3/task1$
```

(2) Task 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.

```
 GNU nano 8.3                          c2.c
#include <stdio.h>
#include <stdlib.h>
#include <dirent.h>
#include <sys/types.h>
int main(int argc, char* argv[])
{
    DIR* dirp;
    struct dirent* direntp;
    if ((dirp = opendir(argv[1])) == NULL) {
        printf("error\n");
        exit(1);
    }
    while ((direntp = readdir(dirp)) != NULL)
        printf("%s\n", direntp->d_name);
    closedir(dirp);
    exit(0);
}
```

```
 GNU nano 8.3                          Makefile
hello2:c2.o
        gcc -o hello2 c2.o
c2.o:c2.c
        gcc -c c2.c
clean:
        rm -rf *.o
```

```
B22040104谢淡恬@ubuntu64:~/exp_3/task2$ nano c2.c
B22040104谢淡恬@ubuntu64:~/exp_3/task2$ nano Makefile
B22040104谢淡恬@ubuntu64:~/exp_3/task2$ make
gcc -c c2.c
gcc -o hello2 c2.o
B22040104谢淡恬@ubuntu64:~/exp_3/task2$ ./hello2 .
..
c2.o
c2.c
Makefile
.
hello2
B22040104谢淡恬@ubuntu64:~/exp_3/task2$ make clean
rm -rf *.o
B22040104谢淡恬@ubuntu64:~/exp_3/task2$
```

(3) Task 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.

```
 GNU nano 8.3                          c3.c
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
int main(){
    char buf[1024] = {0};
    char buf2[1024] = {0};
    getcwd(buf, 1024);
    printf("%s\n", buf);
    if(chdir("/home")<0){
        printf("error\n");
    }
    else
    {
        printf("success\n");
    }
    getcwd(buf2,1024);
    printf("%s\n",buf2);
    return 0;
}
```

```
 GNU nano 8.3                        Makefile
hello3:c3.o
        gcc -o hello3 c3.o
c3.o:c3.c
        gcc -c c3.c
clean:
        rm -rf *.o
```

```
B22040104谢淡恬@ubuntu64:~/exp_3/task3$ nano c3.c
B22040104谢淡恬@ubuntu64:~/exp_3/task3$ nano Makefile
B22040104谢淡恬@ubuntu64:~/exp_3/task3$ make
gcc -c c3.c
gcc -o hello3 c3.o
B22040104谢淡恬@ubuntu64:~/exp_3/task3$ ./hello3
/home/B22040104谢淡恬/exp_3/task3
success
/home
B22040104谢淡恬@ubuntu64:~/exp_3/task3$ make clean
rm -rf *.o
B22040104谢淡恬@ubuntu64:~/exp_3/task3$
```