

南京邮电大学

# 实验报告

(2024 / 2025 学年 第一 学期)

课程名称	Linux 编程			
实验名称	实验三			
实验时间	2024	年	12	月 13 日
指导单位	计算机学院、软件学院、网络空间安全学院			
指导教师	王磊			

学生姓名	田伟豪	班级学号	B22041015
学院(系)	计算机学院	专 业	信息安全

## 实 验 报 告

实验名称	实验三			指导教师	王磊
实验类型	上机	实验学时	4	实验时间	2024.12.13
<p>一、 实验目的和要求</p> <p>通过实验熟悉 Linux 操作系统环境，掌握基本的 Linux 命令使用</p>					
<p>二、实验环境(实验设备)</p> <p>Win11+Vmware Workstation Pro+Ubuntu</p>					
<p>三、实验原理及内容</p> <p>进一步在 Linux 系统中使用 C 编程语言的基本语法，加深对其认知。</p>					

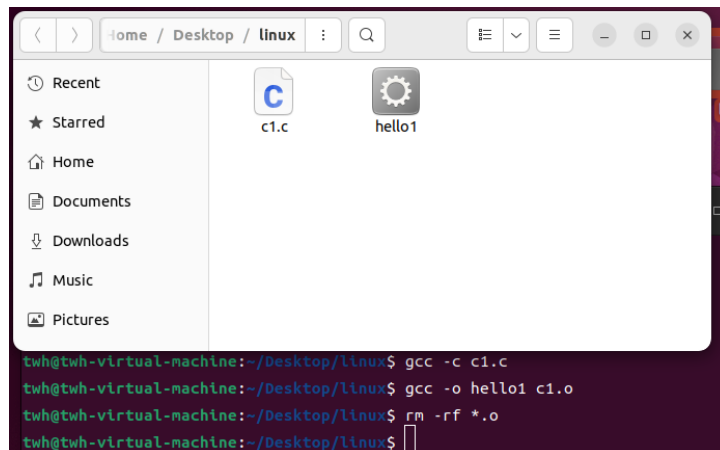
## 实验报告

(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.

实验源码:

```
#include <stdio.h> // 包含标准输入输出库
int main(int argc, char* argv[]) {
    // 定义一个字符数组 buf, 用于存储读取的文件内容, 初始化为全零
    char buf[1024] = { 0 };
    // 检查命令行参数个数是否小于 2, 如果小于 2, 提示用户输入源文件
    if (argc < 2){
        printf("please input source file!\n");
        return -1; // 返回-1 表示程序错误退出
    }
    // 尝试以只读方式打开用户提供的文件, 文件名从命令行参数 argv[1]获取
    FILE* fp = fopen(argv[1], "r");
    // 如果文件打开失败, 打印错误信息并返回-1
    if (fp == NULL) {
        printf("open source %s failed\n", argv[1]);
        return -1;
    }
    // 使用 fgets 函数逐行读取文件内容, 每次最多读取 1024 字节存储到 buf 中
    while (fgets(buf, 1024, fp)){
        // 打印读取到的每一行内容
        printf("%s\n", buf);
    }
    // 程序成功执行完毕, 返回 0 表示正常退出
    return 0;
}
```

运行结果:



# 实验报告

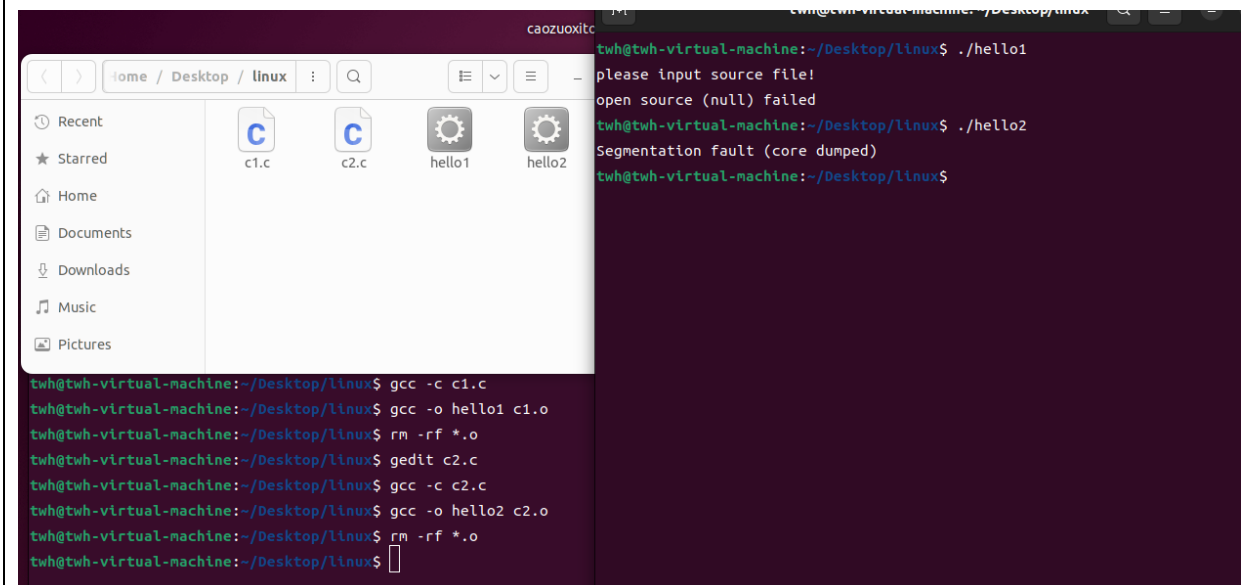
(2) Write a C program that displays all the file names in the current directory. The program is compiled and linked by the make tool, which requires the generation of the.o file first, and then the generation of the executable file, and the function of deleting the intermediate file (.o) in the makefile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

### 实验源码:

```
#include <stdio.h>
#include <dirent.h>

int main(int argc, char* argv[]){
    DIR* dirp; // 定义 DIR 类型的指针，用于表示目录流
    struct dirent* direntp; // 定义指向 dirent 结构体的指针，用于存储读取到的目录项
    // 尝试打开由命令行参数 argv[1]指定的目录
    if ((dirp = opendir(argv[1])) == NULL) {
        // 如果目录打开失败，打印错误信息
        printf("error\n");
        return -1; // 返回-1 表示程序错误退出
    }
    // 使用 readdir 函数逐个读取目录项，直到读取完目录中的所有项
    while ((direntp = readdir(dirp)) != NULL)
        // 打印读取到的每个目录项的名称
        printf("%s\n", direntp->d_name);
    // 关闭目录流
    closedir(dirp);
    // 程序成功执行完毕，返回 0 表示正常退出
    return 0;
}
```

运行结果:



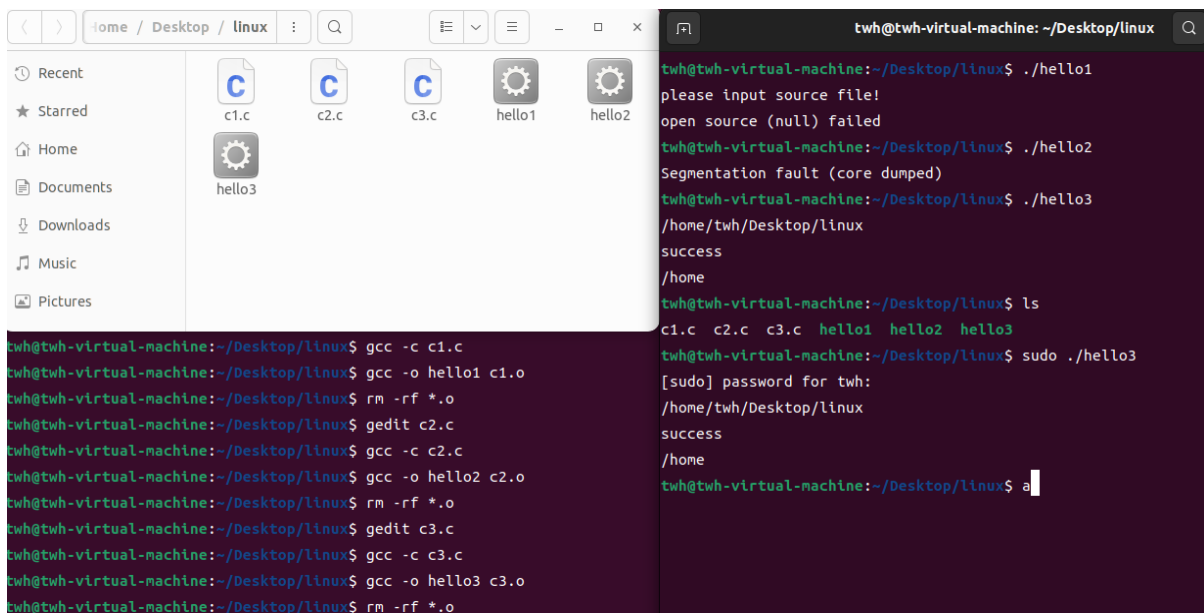
## 实验报告

(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.

源代码:

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
int main() {
    // 定义两个字符数组 buf 和 buf2, 大小为 1024 字节, 用于存储当前工作目录路径
    char buf[1024] = {0};
    char buf2[1024] = {0};
    // 获取当前工作目录并存储到 buf 中
    getcwd(buf, 1024);
    printf("%s\n", buf);
    // 尝试将当前工作目录更改为"/home"
    if (chdir("/home") < 0) {
        printf("error\n"); // 如果更改目录失败, 打印错误信息
    } else {
        printf("success\n"); // 如果更改目录成功, 打印成功信息
    }
    getcwd(buf2, 1024); // 获取更改后的当前工作目录并存储到 buf2 中
    printf("%s\n", buf2); // 打印更改后的当前工作目录
    return 0;
}
```

运行结果:



```
twh@twh-virtual-machine: ~/Desktop/linux
Recent
Starred
Home
Documents
Downloads
Music
Pictures

c1.c
c2.c
c3.c
hello1
hello2
hello3

twh@twh-virtual-machine:~/Desktop/linux$ ./hello1
please input source file!
open source (null) failed
twh@twh-virtual-machine:~/Desktop/linux$ ./hello2
Segmentation fault (core dumped)
twh@twh-virtual-machine:~/Desktop/linux$ ./hello3
/home/twh/Desktop/linux
success
/home
twh@twh-virtual-machine:~/Desktop/linux$ ls
c1.c c2.c c3.c hello1 hello2 hello3
twh@twh-virtual-machine:~/Desktop/linux$ sudo ./hello3
[sudo] password for twh:
/home/twh/Desktop/linux
success
/home
twh@twh-virtual-machine:~/Desktop/linux$ a
```

```
twh@twh-virtual-machine:~/Desktop/linux$ gcc -c c1.c
twh@twh-virtual-machine:~/Desktop/linux$ gcc -o hello1 c1.o
twh@twh-virtual-machine:~/Desktop/linux$ rm -rf *.o
twh@twh-virtual-machine:~/Desktop/linux$ gedit c2.c
twh@twh-virtual-machine:~/Desktop/linux$ gcc -c c2.c
twh@twh-virtual-machine:~/Desktop/linux$ gcc -o hello2 c2.o
twh@twh-virtual-machine:~/Desktop/linux$ rm -rf *.o
twh@twh-virtual-machine:~/Desktop/linux$ gedit c3.c
twh@twh-virtual-machine:~/Desktop/linux$ gcc -c c3.c
twh@twh-virtual-machine:~/Desktop/linux$ gcc -o hello3 c3.o
twh@twh-virtual-machine:~/Desktop/linux$ rm -rf *.o
```

## 实 验 报 告

### 四、实验小结（包括总结上机调试过程中所遇到的问题和解决方法、感想与建议等）

通过本节实验课的学习，我加强了对与 linux 环境下 c 语言程序的编译与链接认知并进行实操，进一步学习了 linux 系统的操作。

### 五、指导教师评语

成 绩		批阅人		日 期	
-----	--	-----	--	-----	--