

Linux 第三次实验

1, 创建一个C程序来显示文本文件内容并使用makefile进行编译、链接和清理中间文件的任务。

c 语言程序:

```
#include <stdio.h>
#include <stdlib.h> // 引入 exit 函数所需的头文件

int main(int argc, char* argv[])
{
    if (argc < 2)
    {
        printf("请输入源文件! \n");
        return -1; // 如果没有提供文件名, 则提前退出
    }

    char buf[1024] = { 0 };
    FILE* fp = fopen(argv[1], "r");
    if (fp == NULL)
    {
        printf("打开源文件 %s 失败\n", argv[1]);
        return -1;
    }

    while (fgets(buf, 1024, fp))
    {
        printf("%s", buf); // 如果 fgets 保留了换行符, 则不需要额外添加换行符
    }

    fclose(fp); // 读取后关闭文件
    return 0;
}
```

Makefile 可执行程序:

```
makefile
# 用于编译 c1.c 并创建 hello1 可执行文件的 makefile

# 从 c1.o 创建可执行文件 hello1 的规则
hello1: c1.o
    gcc -o hello1 c1.o

# 从源文件 c1.c 创建目标文件 c1.o 的规则
c1.o: c1.c
```

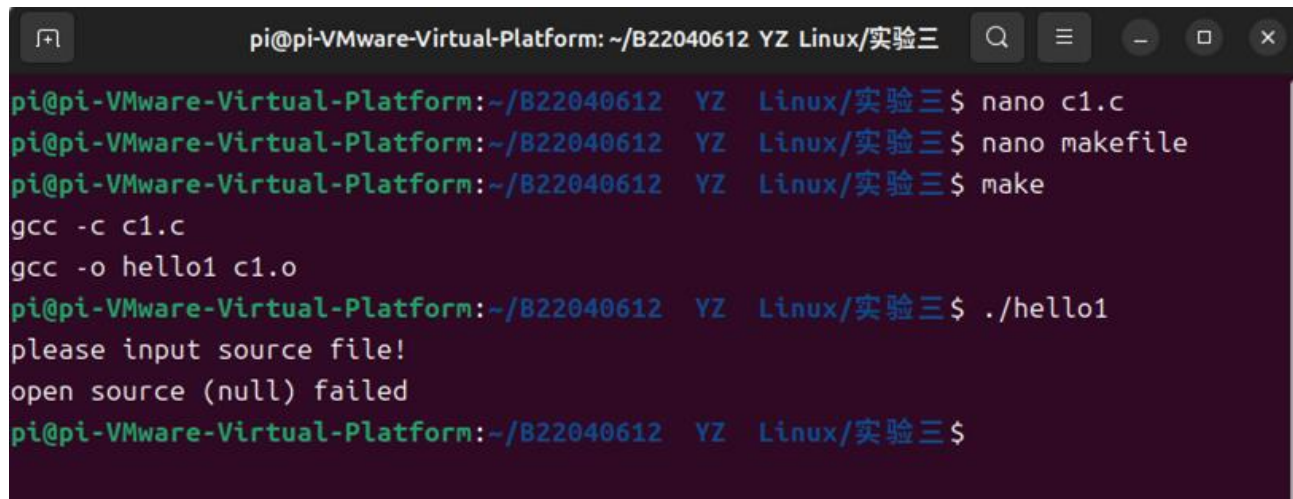
```
gcc -c c1.c
```

清理中间目标文件的规则

clean:

```
rm -f *.o hello1
```

运行结果:



```
pi@pi-VMware-Virtual-Platform: ~/B22040612 YZ Linux/实验三
pi@pi-VMware-Virtual-Platform:~/B22040612 YZ Linux/实验三$ nano c1.c
pi@pi-VMware-Virtual-Platform:~/B22040612 YZ Linux/实验三$ nano makefile
pi@pi-VMware-Virtual-Platform:~/B22040612 YZ Linux/实验三$ make
gcc -c c1.c
gcc -o hello1 c1.o
pi@pi-VMware-Virtual-Platform:~/B22040612 YZ Linux/实验三$ ./hello1
please input source file!
open source (null) failed
pi@pi-VMware-Virtual-Platform:~/B22040612 YZ Linux/实验三$
```

2, 创建一个C程序来显示文本文件内容并使用makefile进行编译、链接和清理中间文件的任务。

c 程序文件:

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

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

```
#include <sys/types.h>
```

```
int main(int argc, char* argv[]) {
```

```
    DIR* dirp;
```

```
    struct dirent* direntp;
```

```
// 如果 argc > 1, 则打开 argv[1]指定的目录; 否则, 打开当前目录
```

```
// 由于要求列出当前目录, 这里我们直接打开"."
```

```
if ((dirp = opendir(".")) == NULL) {
```

```
    perror("opendir"); // 使用 perror 打印更具体的错误信息
```

```
    return 1; // 使用 return 而不是注释掉的 exit
```

```
}
```

```
while ((direntp = readdir(dirp)) != NULL) {
```

```
    printf("%s\n", direntp->d_name);
```

```
}
```

```
closedir(dirp);
```

```
return 0; // 使用 return 而不是注释掉的 exit
```

```
}
```

makefile 程序文件：

makefile

hello2: c2.o

gcc -o hello2 c2.o

c2.o: c2.c

gcc -c c2.c

clean:

rm -f *.o hello2

运行结果：



```
pi@pi-VMware-Virtual-Platform: ~/B22040612 YZ Linux/实验三/2$ nano c2.c
pi@pi-VMware-Virtual-Platform:~/B22040612 YZ Linux/实验三/2$ nano makefile
pi@pi-VMware-Virtual-Platform:~/B22040612 YZ Linux/实验三/2$ make
gcc -c c2.c
gcc -o hello2 c2.o
pi@pi-VMware-Virtual-Platform:~/B22040612 YZ Linux/实验三/2$ ./hello2
hello2
c2.o
..
makefile
.
c2.c
pi@pi-VMware-Virtual-Platform:~/B22040612 YZ Linux/实验三/2$
```

3,创建一个C 程序来显示文本文件内容并使用makefile 进行编译、链接和清理中间文件的任务。

c 程序文件：

#include <stdio.h>

#include <stdlib.h>

#include <unistd.h>

int main() {

char buf[1024] = {0};

char buf2[1024] = {0};

getcwd(buf, sizeof(buf));

printf("Current directory: %s\n", buf);

if (chdir("/home") < 0) {

perror("chdir"); // 使用 perror 打印错误信息

return 1; // 返回非零值表示错误

} else {

```
    printf("Changed directory successfully.\n");  
}
```

```
getcwd(buf2, sizeof(buf2));  
printf("New current directory: %s\n", buf2);
```

```
    return 0; // 返回零表示成功  
}
```

makefile 程序文件：

makefile

hello3: c3.o

gcc -o hello3 c3.o # 更正了可执行文件的名称

c3.o: c3.c

gcc -c c3.c

clean:

rm -f *.o hello3 # 删除了所有相关的构建文件

运行结果：



```
pi@pi-VMware-Virtual-Platform: ~/B22040612 YZ Linux/实验三/3  
pi@pi-VMware-Virtual-Platform:~/B22040612 YZ Linux/实验三/3$ nano c3.c  
pi@pi-VMware-Virtual-Platform:~/B22040612 YZ Linux/实验三/3$ nano makefile  
pi@pi-VMware-Virtual-Platform:~/B22040612 YZ Linux/实验三/3$ make  
gcc -c c3.c  
gcc -o hello3 c3.o # 更正了可执行文件的名称  
pi@pi-VMware-Virtual-Platform:~/B22040612 YZ Linux/实验三/3$ ./hello3  
Current directory: /home/pi/B22040612 YZ Linux/实验三/3  
Changed directory successfully.  
New current directory: /home  
pi@pi-VMware-Virtual-Platform:~/B22040612 YZ Linux/实验三/3$
```