

# Experiment3 C programming experiment

## (1) Task 1

```
B23090326@k-virtual-machine:~/course/linux_编程/实验3$ gedit c1.c
B23090326@k-virtual-machine:~/course/linux_编程/实验3$ gedit Makefile
B23090326@k-virtual-machine:~/course/linux_编程/实验3$ make
Makefile:2: *** 遗漏分隔符 (null)。 停止。
B23090326@k-virtual-machine:~/course/linux_编程/实验3$ make
Makefile:2: *** 遗漏分隔符 (null)。 停止。
B23090326@k-virtual-machine:~/course/linux_编程/实验3$ make
gcc -c c1.c
gcc -o hello1 c1.o
B23090326@k-virtual-machine:~/course/linux_编程/实验3$ ls
c1.c c1.o hello1 Makefile
B23090326@k-virtual-machine:~/course/linux_编程/实验3$ ./hello1 test1.txt
open source test1.txt failed
B23090326@k-virtual-machine:~/course/linux_编程/实验3$ touch test1.txt
B23090326@k-virtual-machine:~/course/linux_编程/实验3$ ./hello1 test1.txt
B23090326@k-virtual-machine:~/course/linux_编程/实验3$ ./hello1 test1.txt
我是B23090326燕康
B23090326@k-virtual-machine:~/course/linux_编程/实验3$ make clean
rm -rf *.o
B23090326@k-virtual-machine:~/course/linux_编程/实验3$ ls
c1.c hello1 Makefile test1.txt
B23090326@k-virtual-machine:~/course/linux_编程/实验3$ █
```

首先编写 c1.c 程序并通过 Makefile 编译生成可执行文件 hello1；创建测试文件 test1.txt 并写入内容，运行程序成功显示文件内容；最后执行 make clean 清理中间文件。

## (2) Task 2

```
修改 c2.c (仅显示部分)
B23090326@k-virtual-machine:~/course/linux_编程/实验3$ gedit c2.c
B23090326@k-virtual-machine:~/course/linux_编程/实验3$ gedit Makefile
B23090326@k-virtual-machine:~/course/linux_编程/实验3$ make
gcc -c c2.c
gcc -o hello1 c2.o
B23090326@k-virtual-machine:~/course/linux_编程/实验3$ ls
c1.c c2.c c2.o hello1 Makefile test1.txt
B23090326@k-virtual-machine:~/course/linux_编程/实验3$ ./hello1 .
.
hello1
test1.txt
Makefile
..
c2.o
c1.c
c2.c
B23090326@k-virtual-machine:~/course/linux_编程/实验3$ make clean
rm -rf *.o
B23090326@k-virtual-machine:~/course/linux_编程/实验3$ ls
c1.c c2.c hello1 Makefile test1.txt
B23090326@k-virtual-machine:~/course/linux_编程/实验3$ █
```

修改 c2.c 程序并重新编译，运行./hello1 . 正确显示当前目录下的所有文件名（包括隐藏文件和子目录标识），验证了目录操作函数的功能；最后使用 make clean 清理中间文件，完整实现了目录内容读取功能。

## (3) Task 3

```
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
B23090326@k-virtual-machine:~/course/linux_编程/实验3$ gedit c3.c
B23090326@k-virtual-machine:~/course/linux_编程/实验3$ gedit Makefile
B23090326@k-virtual-machine:~/course/linux_编程/实验3$ make
gcc -c c3.c
gcc -o hello1 c3.o
B23090326@k-virtual-machine:~/course/linux_编程/实验3$ ls
c1.c  c2.c  c3.c  c3.o  hello1  Makefile  test1.txt
B23090326@k-virtual-machine:~/course/linux_编程/实验3$ ./hello1
/home/B23090326/course/linux_编程/实验3
success
/home
B23090326@k-virtual-machine:~/course/linux_编程/实验3$ make clean
rm -rf *.o
B23090326@k-virtual-machine:~/course/linux_编程/实验3$ ls
c1.c  c2.c  c3.c  hello1  Makefile  test1.txt
B23090326@k-virtual-machine:~/course/linux_编程/实验3$
```

程序正确获取当前工作目录并显示路径，成功切换到/home 目录后再次显示新目录路径，验证了 chdir() 函数改变进程工作目录的功能；最后执行 make clean 清理中间文件，完整实现了目录切换功能。

#### (4) 问题与解决：

##### (1) makefile 格式缩进错误

我们遇到了一个典型的 Makefile 语法错误。当直接复制粘贴实验指导中提供的 Makefile 代码并执行 make 命令时，系统报错“Makefile:2: \*\*\* 遗漏分隔符 (null)。停止。”。经过排查，发现该错误是由于 Makefile 中规则命令行的缩进格式不正确所致。Makefile 要求所有规则下的命令行必须使用 Tab 字符进行缩进，而复制粘贴的代码可能使用了空格或其他空白字符，导致 make 工具无法正确解析。

```
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
B23090326@k-virtual-machine:~/course/linux_编程/实验3$ gedit c1.c
B23090326@k-virtual-machine:~/course/linux_编程/实验3$ gedit Makefile
B23090326@k-virtual-machine:~/course/linux_编程/实验3$ make
Makefile:2: *** 遗漏分隔符 (null)。停止。
B23090326@k-virtual-machine:~/course/linux_编程/实验3$ make
Makefile:2: *** 遗漏分隔符 (null)。停止。
B23090326@k-virtual-machine:~/course/linux_编程/实验3$
```

我们通过手动编辑 Makefile 文件，将命令行前的缩进统一改为 Tab 键，并保存修改。随后重新运行 make 命令，编译和链接过程顺利执行，成功生成了目标可执行文件。这一问题的解决不仅保证了实验任务的顺利完成，也加深了我们对 Makefile 语法规则的理解，特别是其对于格式的严格性要求。在今后的实验和项目中，我们将更加注意 Makefile 的书写规范，避免类似错误的发生。

```
打开(O) ▾ B23090326燕康
hello1: c1.o
        gcc -o hello1 c1.o
c1.o: c1.c
        gcc -c c1.c
clean:
        rm -rf *.o
```

