**广州大学学生实验报告**

开课学院及实验室: 计算机科学与网络工程学院 2024年 5月 24日

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 学院 | 计算机科学与网络工程学院 | 年级/专业/班 |  | 姓名 |  | 学号 |  |
| 实验课程名称 | 操作系统实验 | | | | | 成绩 |  |
| 实验项目名称 | 文件系统 | | | | | 指导老师 |  |

**实验四 文件系统**

1. **实验目的**

**1、熟悉Linux文件系统的文件和目录结构，掌握Linux文件系统的基本特征；**

**2、模拟实现Linux文件系统的简单I/O流操作：备份文件。**

**二、基本要求**

**1、对文件系统命令和库函数要提前熟悉；**

**2、画出备份文件的过程图；**

**3、测试数据文件提前准备好；**

**4、编写程序并调试；**

**5、截屏输出实验结果；**

**6、根据实验结果与理论课讲述的原理进行实验分析。**

**三、实验软件**

**Ubuntu、gcc编译器、Linux操作系统**

**四、实验内容：**

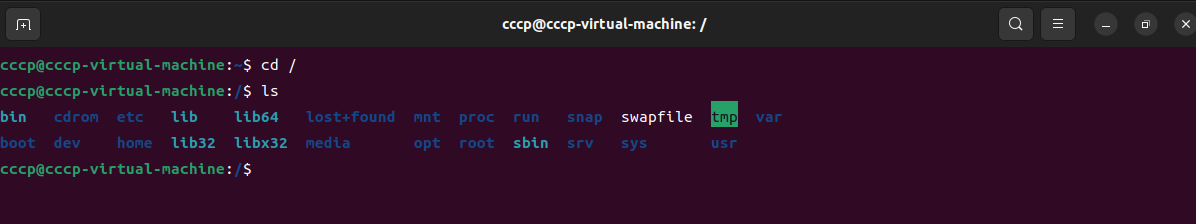
**1.浏览Linux系统根目录下的子目录，熟悉每个目录的文件和功能。**

**实验原理:**

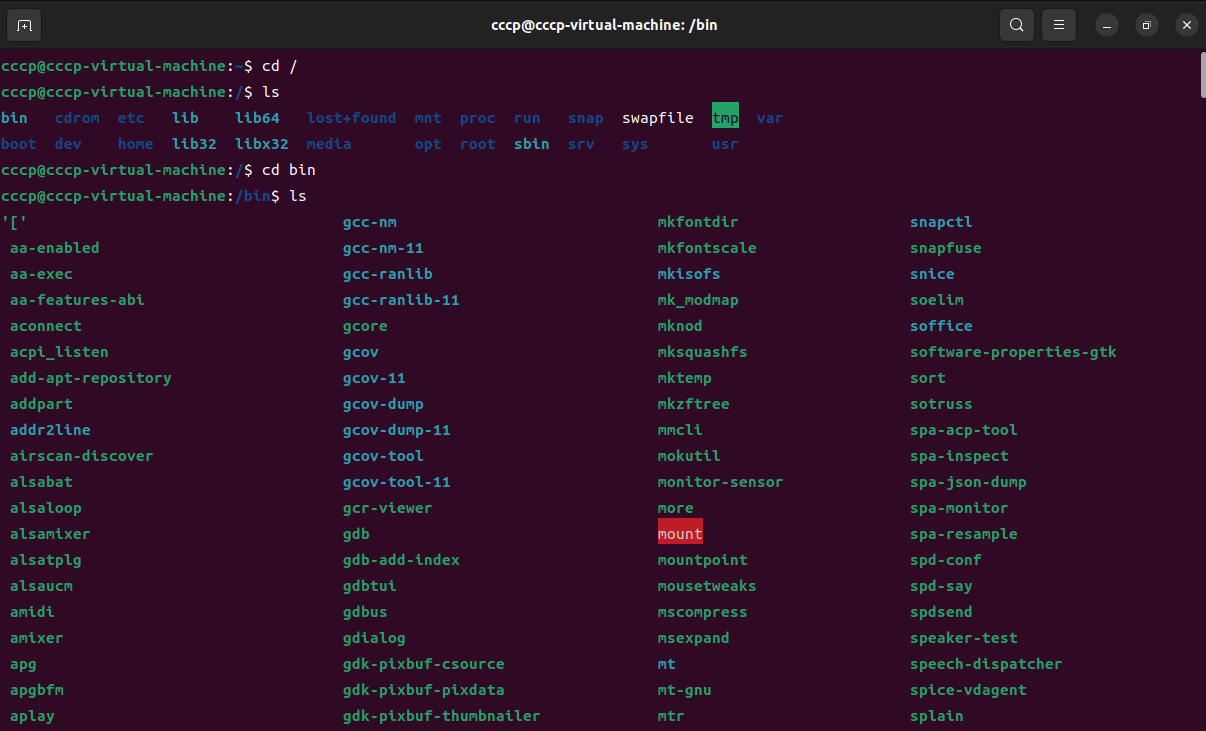
**cd命令：用来切换当前目录至别的地方。例如，cd /表示进入系统根目录。cd ../表示进行目录回退。**

**Linux系统下的根目录下的子目录包括:bin-二进制应用程序、boot-启动配置文件、dev-设备文件包含设备文件、etc-配置文件、启动脚本等、home-本地用户主目录、lib-系统库文件、lost+found-根目录下提供一个遗失+查找系统。**

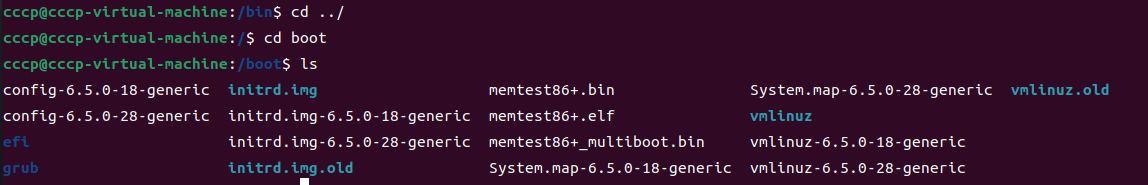
**进入根系统目录:**

****

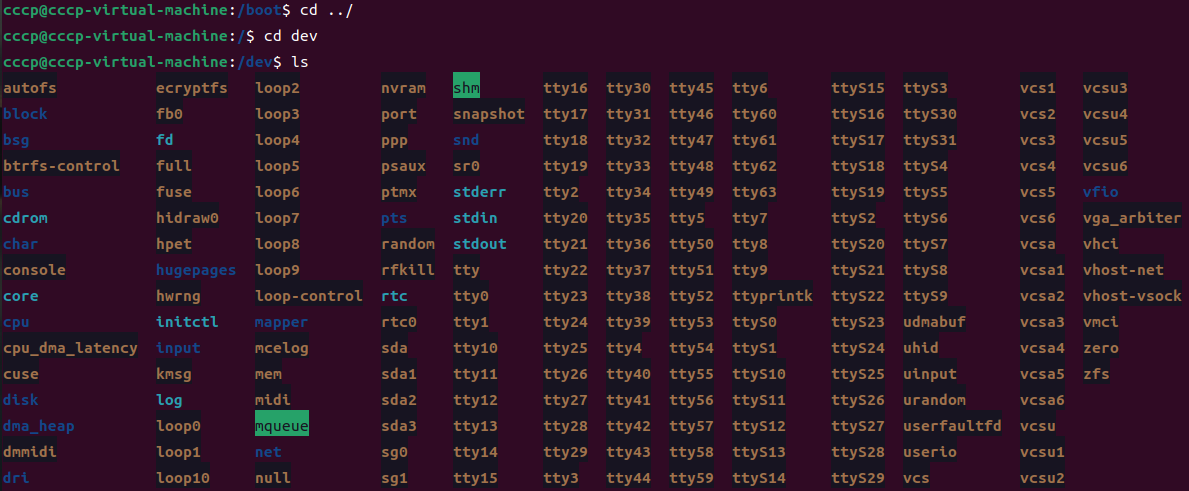
**进入文件bin:**

****

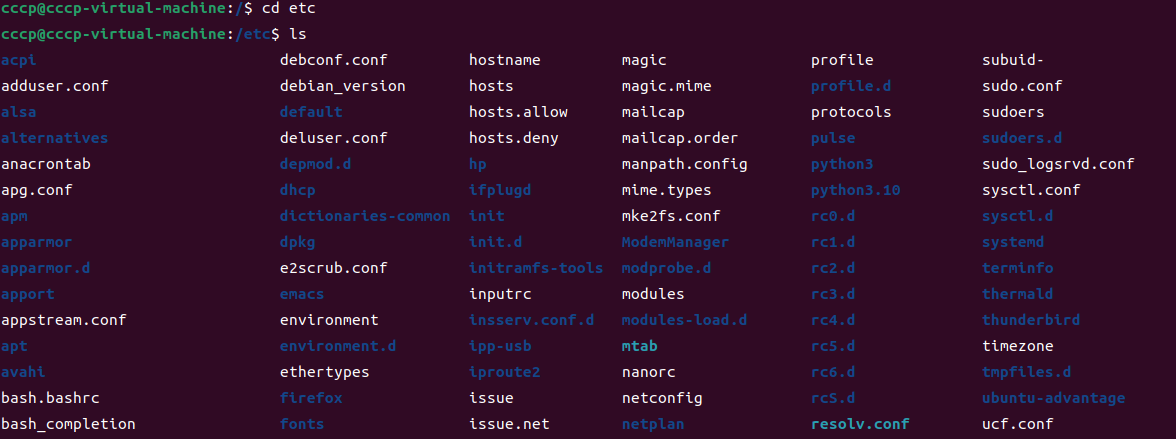
**进入文件boot:**

****

**进入文件dev:**

****

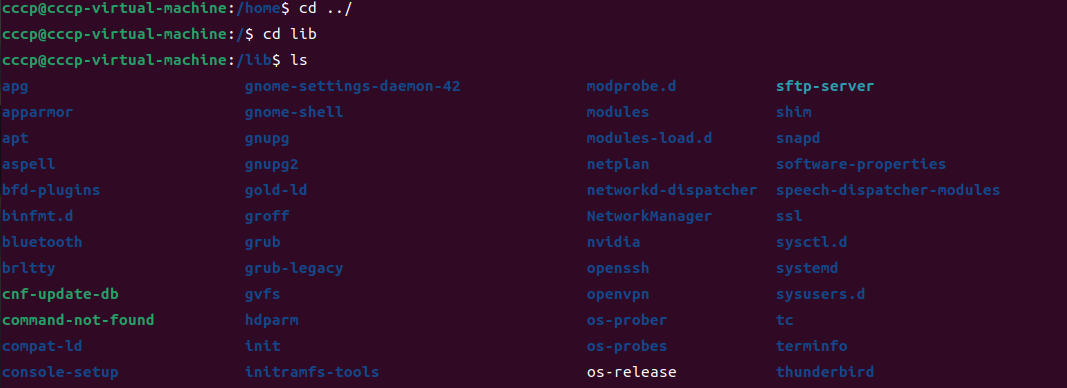
**进入文件ect:**

****

**进入文件home:**

****

**进入文件lib:**

****

**进入lost+found文件:**

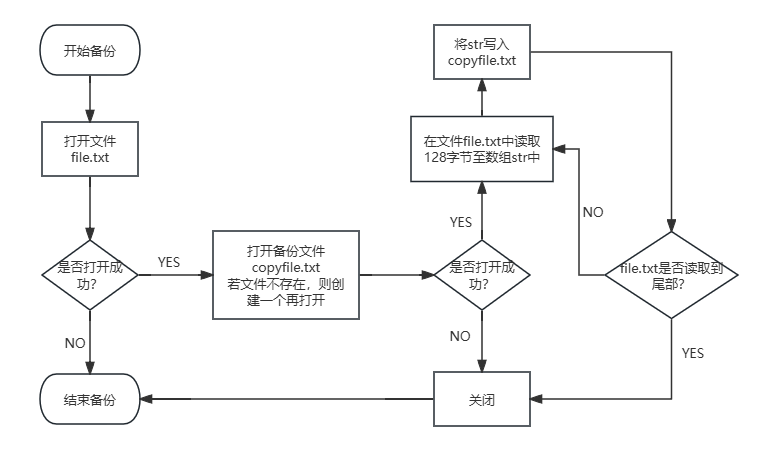
**若直接使用cd lost+found命令，终端将会提示权限不足。因为lost+found是存储发生意外后丢失的文件的，只有根用户才能打开。**

**想要打开该文件先执行命令sudo** **chmod -R 777 lost+found，再使用cd lost+found命令。**

****

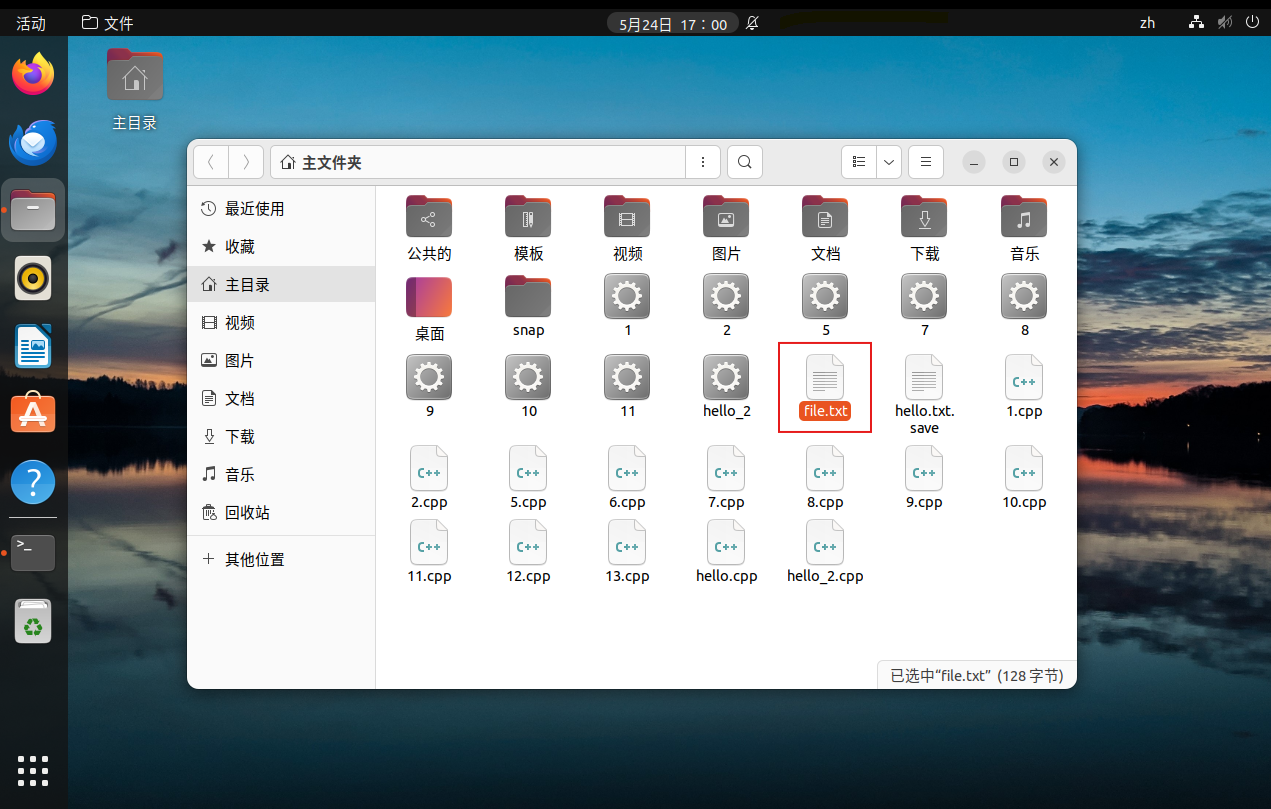
**2.设计程序模拟实现Linux文件系统的简单I/O流操作：备份文件。**

**实验流程图:**

****

**测试数据及文件:**

**测试文件路径为/home/cccp/file.txt**

****

**测试文件内容:**

****

**实现代码:**

**Buff缓冲区:**

char str[128];

**使用C库函数实现文件备份:**

//采用C库函数实现文件备份

void C\_copy()

{

    FILE\* p1 = fopen("/home/cccp/file.txt", "r");//打开文件

    //如果打开失败

    if (p1 == NULL)

    {

        cout << "文件打开失败!" << endl;

        return;

    }

    else

    {

        //打开备份目标文件,若不存在则创建一个

        FILE\* p2 = fopen("/home/cccp/copyfile\_1.txt", "w+");

        if (p2 == NULL)//打开失败

        {

            cout << "文件打开失败!" << endl;

            fclose(p1);//关闭文件file.txt

            return;

        }

        else//成功

        {

            //在file中读取到str中，再copy到copyfile中

            int size;

            while (size = fread(str, sizeof(char), 128, p1))//fread返回一个int值

            {

                fwrite(str, sizeof(char), size, p2);//写入文件

            }

        }

        fclose(p1);//关闭文件

        fclose(p2);

        cout << "文件备份成功" << endl;

    }

    return;

}

**使用系统调用实现文件备份:**

//采用系统调用实现文件备份

void system\_copy()

{

    int p1, p2;

    p1 = open("/home/cccp/file.txt", O\_RDONLY);//打开文件

    if (p1==-1)//若文件打开失败

    {

        cout << "文件打开失败!" << endl;

        return;

    }

    else

    {

        p2 = open("/home/cccp/copyfile\_2.txt", O\_WRONLY|O\_CREAT);//打开文件

        if (p2 == -1)

        {

            cout << "文件打开失败!" << endl;

            close(p1);//关闭文件

            return;

        }

        else

        {

            int size;

            while (size = read(p1, str, sizeof(str)))//读入

            {

                write(p2, str, sizeof(str));//写

            }

            close(p1);//关闭文件

            close(p2);

            cout << "文件备份完成" << endl;

        }

    }

    return;

}

**主函数:**

int main()

{

    //C库函数

    C\_copy();

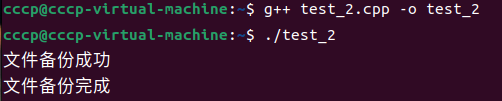
    //系统调用函数

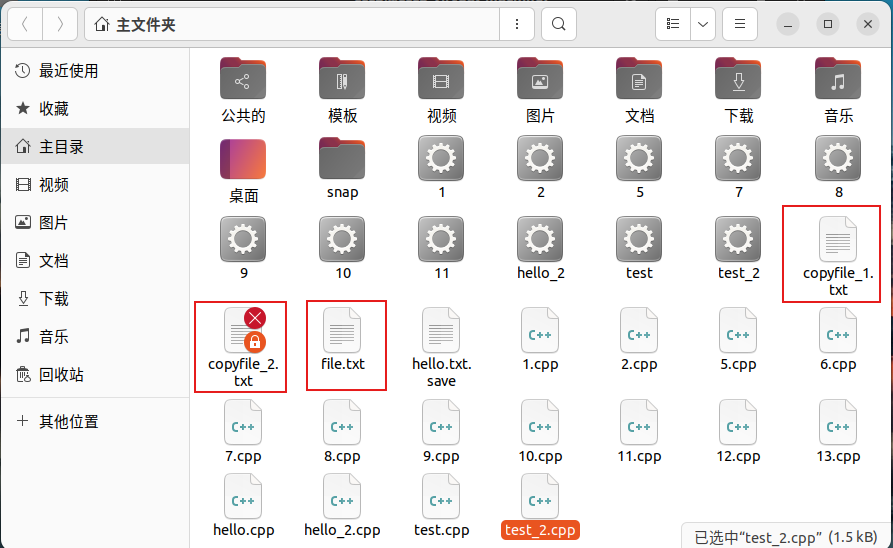
    system\_copy();

    return 0;

}

**调试结果:**

****

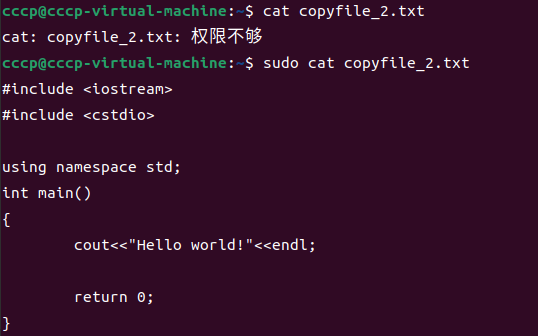
****

**上图中文件分别为使用C库函数实现备份文件copyfile\_1.txt何使用系统调用实现备份的copyfile\_2.txt。file.txt为原文件。**

**下图是copyfile\_1内容:**

****

**直接打开copyfile\_2会显示权限不足，因此打开方式如下图所示:**

****

**五、实验思考**

**1、使用系统调用函数open(),read(),write(),close()实现简单文件备份的原理是什么？**

**答:**

**在使用系统调用open()函数时，需要指定一个文件路径和对文件的操作方式，如int p = open(“文件路径”,操作方式1|操作方式2|操作方式3)。操作方式包含:** **O\_RDONLY只读打开，O\_WRONLY只写打开，O\_RDWR读/写打开。打开失败返回-1。**

**在系统调用中使用read()函数时，int size = read(文件int p,缓冲器,读取数据大小)。将读取到的数据存储到缓冲器中，读取失败时返回-1，成功时返回读取字节大小。**

**在使用系统调用write()函数时，write(文件int p,缓冲器,写入数据大小)。当写入失败时，返回-1。**

**在使用系统调用close(),关闭文件。**

**2、使用C库函数fopen(), fread(), fwrite(), fclose() 来实现简单文件备份的原理是什么？**

**答:**

**fopen()，可以实现以指定的文件路径打开文件，并进行读取操作，同时可以将文件备份到指定的备份文件中。**

**fread()函数是C标准库中提供的文件读取函数，可以从指定的文件流中读取指定数量的数据，并将其存储到指定的缓冲区中。**

**fwrite()函数是C标准库中提供的文件写入函数，可以将指定缓冲区中的数据写入到指定的文件流中，写入数据的大小由参数size和nmemb指定。**

**fclose()函数是C标准库中提供的文件流关闭函数，可以关闭指定的文件流，并释放相关的资源。**

**具体使用方法代码中有。**

**3、上述二者区别在哪里？**

**答:**

**(1)系统调用函数需要进入内核态执行函数,需要内核态与用户态来回切换,执行效率较低。**

**(2)C库函数不需要内核态与用户态来回切换,执行效率高。**

**(3)** **系统调用函数的具体实现方式在不同的操作系统中可能会有所不同，因此这些函数在不同的操作系统上的移植性较差。而C库函数是标准化的函数库，这些函数在不同的操作系统中都有相同的实现方式和接口规范，因此具有很好的移植性。**