

山东大学 计算机科学与技术 学院

计算机系统原理 课程实验报告

学号：202200400053	姓名：王宇涵	班级：2 班
实验题目： 2.1 C 语言编程——字符操作 2.2 文件读写及加解密		
实验学时：2	实验日期：2023-11-01	
实验目的： 2.1: (1) 了解数据在计算机内的编码表示，熟悉常见字符的 16 进制 ASCII 码； (2) 初步掌握 C 语言，能够用 C 语言进行简单编程实验，掌握键盘输入及显示输出的方法； (3) 熟悉 Linux 环境，学会如何在 Linux 环境下完成 C 程序的编写和运行。 2.2: 以读写文件为例，掌握计算机系统中程序的执行流程。		
硬件环境： 多路处理器计算机教学实验系统是由四片四核龙芯 3A 处理器构成的 16 核 CC-NUMA 结构、内可配置外可扩展结构的实验硬件平台。实验系统特点如下： (1) 多种并行层次：多发射、多核、多路、多机 (2) 多种互连方式：片上网络、HyperTransport、以太网 (3) 多种存储结构：CMP/SMP、CC-NUMA (4) 多种编程模式：Pthread、MPI、OpenMP 该实验系统是由多路处理单元和前端控制单元组成，多路处理单元上有对称的 4 个计算节点，每个计算节点包含一颗龙芯 3A 四核处理器。4 个计算节点既可通过网络互连为多处理机集群架构，也可通过 HT 总线互连为 CC-NUMA 架构。前端控制单元由龙芯 2H 构成，为计算节点提供内核和网络文件系统。		
软件环境： Vmware WorkStation linux 虚拟机		
实验步骤与内容： 2.1: 1. 在 windows 平台上下载安装 VmwareWorkStation 和 linux 虚拟机 2. 下载安装 gcc 3. 进入 linux 终端通过 vi 指令进入 vim 编辑器页面,写 helloworld.c 文件 4. gcc 命令对 helloworld.c 文件进行编译.生成可执行文件 helloworld 5. 进入文件 helloworld,输入相应字符串验证正确性 2.2: 1. 完成 2.1 第 1,2 步 2. 通过 vi 指令进入 vim 编辑器页面,写 origin.txt 文件,输入加密前的源内容 3. 通过 vi 指令进入 vim 编辑器页面,写 code.c 文件,输入有着加密功能的 c 文件		

加密方法:将字符串中每一个字符与'I'm a sdu student'的从左到右的每个字符进行异或

4. 通过 vi 指令进入 vim 编辑器页面,写 decode.c 文件,输入有着解密功能的 c 文件

解密方法:将字符串中每一个字符与'I'm a sdu student'的从左到右的每个字符进行异或

5. 通过 gcc 命令对 code.c 文件进行编译,生成可执行文件 code

6 通过 gcc 命令对 dede.c 文件进行编译,生成可执行文件 decode

7 运行 code 和 decode 文件

结论分析与体会:

本次实验我初次接触了虚拟机和龙芯硬件平台,花了一定时间在虚拟机的安装上,也体会到一个电脑可以同时有两个操作系统.我学习操作 linux 系统中命令行中的文件,并学习了通过 vim 编辑器来进行文件的书写,感到新奇和有趣,同时也通过 C 语言中对字符串的操作,对文件的读写操作成功完成了实验的需求,在 linux 虚拟机终端中成功运行,得到了满意的结果.

这次的实验让我熟悉了 linux 环境,并复习了 C 语言对于文件的读写操作,掌握了计算机系统中程序的执行流程.

附源代码

2.1

Hello world.c

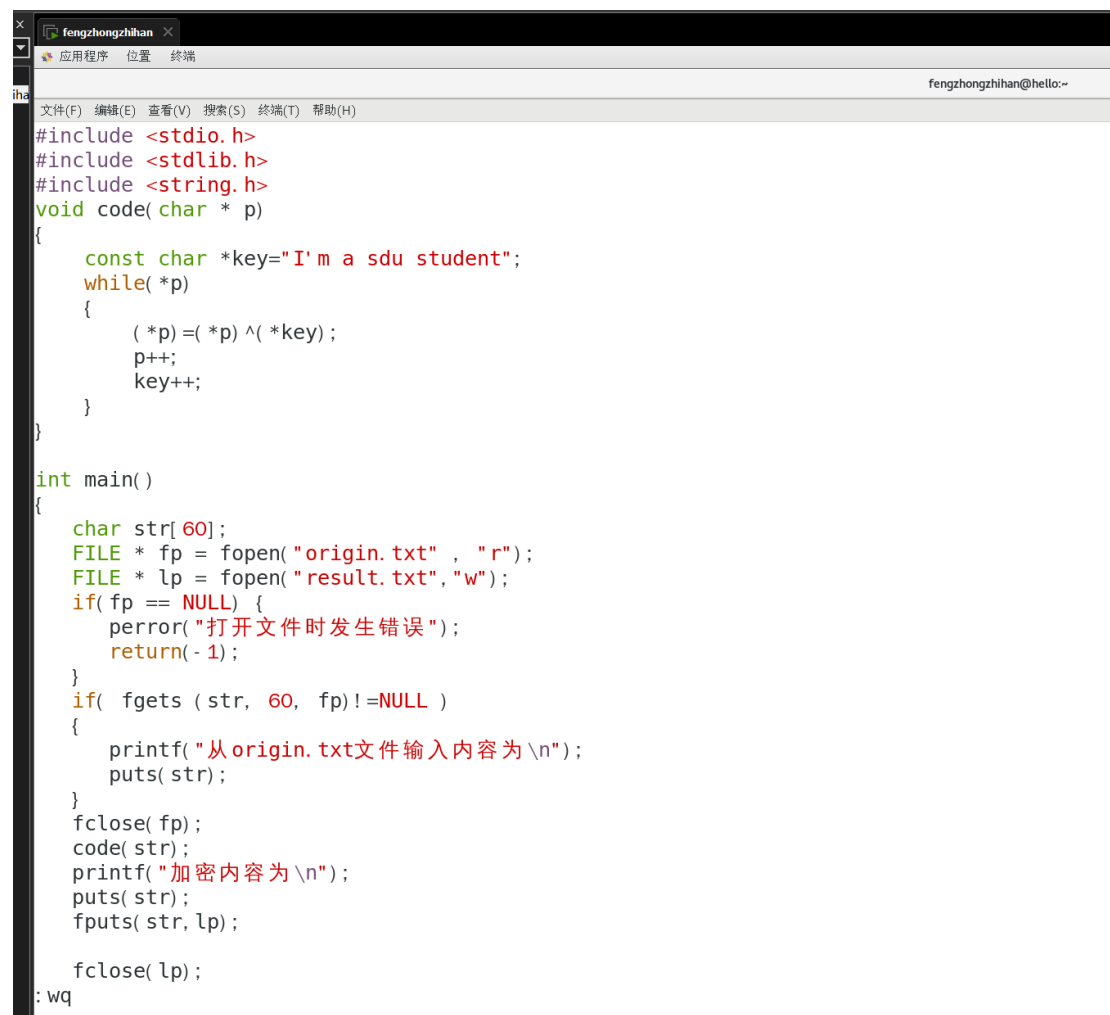
```
fengzhongzhihan@hello:~  
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)  
  
#include<string.h>  
int main()  
{  
    char str[1000];  
    scanf("%s", str);  
    int len=strlen(str);  
    for(int i=0; i<len; i++)  
    {  
        char c=str[i];  
        printf("%x ", c);  
        if(c>='a' && c<='z')  
            printf("%c ", c-32);  
        if(c>='A' && c<='Z')  
            printf("%c ", c+32);  
  
        printf("\n");  
    }  
  
    return 0;  
}  
~  
~  
-- 插入 --  
1, 1 全部
```

测试

```
[ fengzhongzhihan@hello ~]$ ./helloworld
1aABc2
31
61 A
41 a
42 b
63 C
32
```

2.2

code.c



```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
void code(char * p)
{
    const char *key="I'm a sdu student";
    while( *p)
    {
        (*p)=( *p) ^( *key);
        p++;
        key++;
    }
}

int main()
{
    char str[ 60];
    FILE * fp = fopen( "origin.txt" , "r");
    FILE * lp = fopen( "result.txt", "w");
    if( fp == NULL) {
        perror( "打开文件时发生错误");
        return( - 1);
    }
    if( fgets ( str, 60, fp)!=NULL )
    {
        printf( "从origin.txt文件输入内容为\n");
        puts( str);
    }
    fclose( fp);
    code( str);
    printf( "加密内容为\n");
    puts( str);
    fputs( str, lp);

    fclose( lp);
: wq
```

Decode.c

```
应用程序 位置 终端
fengzhongzhihan@hello:~$
#include<stdio.h>
#include <stdlib.h>
#include <string.h>
void decode( char * p)
{
    const char *key="I'm a sdu student";
    while( *p)
    {
        (*p)=( *p) ^( *key);
        p++;
        key++;
    }
}

int main()
{
    FILE *fp;
    char str[ 60];
    /* 打开用于读取的文件 */
    fp = fopen( "result.txt" , "r");
    if( fp == NULL) {
        perror( "打开文件时发生错误");
        return(- 1);
    }
    if( fgets ( str, 60, fp)!=NULL )
    {
        /* 向标准输出 stdout 写入内容 */
        printf( "从result.txt文件中读取加密内容为:\n");
        puts( str);
    }
    fclose( fp);
    printf( "解密内容为:");
    decode( str);
    puts( str);
    return( 0);
}
"decode.c" 37L, 678C
```

Origin.txt

```
fengzhongzhihan x
应用程序 位置 终端
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T)
helloworld
~
~
~
```

测试

```
[ fengzhongzhihan@hello ~]$ gcc code.c -o code
[ fengzhongzhihan@hello ~]$ ./code
从origin.txt文件输入内容为
helloworld

加密内容为
! B LW Dy
加密成功,文件已保存到result.txt文件[ fengzhongzhihan@hello ~]$
```

```
[ fengzhongzhihan@hello ~]$ gcc decode.c -o decode
[ fengzhongzhihan@hello ~]$ ./decode
从 result.txt文件中读取加密内容为:
! B LW    Dy
解密内容为: helloworld
```