西安交通大学实验报告

课程 计算机程序设计 实验名称 第五次纸本作业 第 1 页 共 17 页

系 别 工科试验班(钱学森实验班) 实 验 日 期 2017 年6月 1 日

专业班级 钱学森62班 组别 无 实 验 报 告 日 期 2017 年6 月 5 日

姓 名 周宇晨 学号 2160405046 报 告 退 发 ( 订正 、 重做 ）

同 组 人 无 教 师 审 批 签 字

一、实验目的

1.巩固类和对象、继承、多态、文件的相关知识。

2.熟练掌握函数及各种命令语句。

3.统筹c++相关知识。

二、实验题目

题目一 定义商品楼类building，一个商品楼由若干相同大小的套房flat，每套房由若干相同大小的房间room（房间都是由长宽高直角边构成），应有构造函数、计算商品楼总面积函数、计算一套房的总面积、计算一个房间面积函数和体积函数、运算符==重载函数。

1.要点分析

声明类；构造析构函数；功能函数。

2.程序源码

#include<iostream>

using namespace std;

class Room

{

private:

int a, b, c;

public:

Room() { a = 0; b = 0; c = 0; }

void init(int aa, int bb, int cc) { a = aa; b = bb; c = cc; }

int S() { return a\*b; }

int V() { return a\*b\*c; }

void show() { cout << "一个屋的长宽高：" << a << " " << b << " " << c << endl; }

bool operator == (Room &x)

{

cout << "测试运算符==重载" << endl;

if (S() == x.S() && V() == x.V()) { cout << "两个房间面积和体积都一样大" << endl; return true; }

else { cout << "两个房间面积和体积不都一样大" << endl; return false; }

}

};

class Flat :public Room

{

private:

int n;

public:

Flat() :Room() { n = 0; }

void init(int aa, int bb, int cc, int nn) { Room::init(aa, bb, cc); n = nn; }

int S() { return Room::S()\*n; }

void show() { Room::show(); cout << "一间房的屋子数:" << n << endl; }

};

class Building :public Flat

{

private:

int m;

public:

Building() :Flat() { m = 0; }

void init(int aa, int bb, int cc, int nn, int mm) { Flat::init(aa, bb, cc, nn); m = mm; }

int S() { return Flat::S()\*m; }

void show() { Flat::show(); cout << "一栋楼的房间数:" << m << endl; }

};

int main()

{

Room room;

Flat flat;

Building building;

cout << "初始化结果：" << endl; building.show();

cout << endl;

cout << "请输入一栋楼的房间数、一间房的屋子数、每间屋子的长宽高" << endl;

int a, b, c, n, m;

cin >> m >> n >> a >> b >> c;

room.init(a, b, c);

flat.init(a, b, c, n);

building.init(a, b, c, n, m);

cout << "每间屋子的面积：" << room.S() << endl;

cout << "每间屋子的体积：" << room.V() << endl;

cout << "每间房的面积：" << flat.S() << endl;

cout << "每栋楼的面积：" << building.S() << endl;

cout << endl;

cout << "请再输入一间屋子的长宽高：" << endl;

int d, e, f;

Room roomm;

cin >> d >> e >> f;

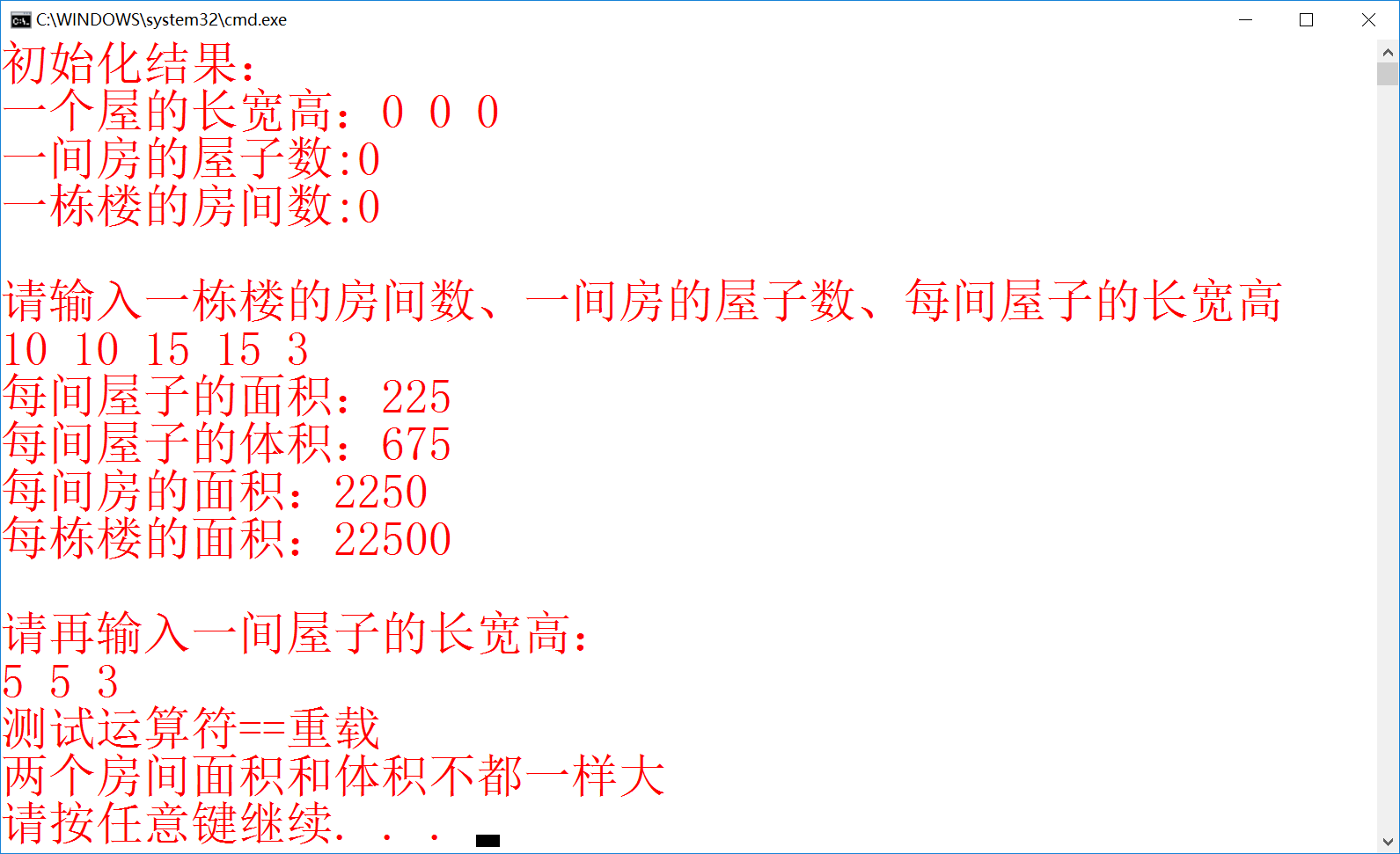
roomm.init(d, e, f);

room == roomm;

return 0;

}

3.实验结果



题目二 定义长方体类cuboid ，应有两个构造函数（即cuboid () 和cuboid (int, int, int) ）、计算体积函数、计算长方体总面积函数 、运算符 + 重载函数（注意：对应长宽高相加，仍然形成长方体），运算符= = 重载函数。

1.要点分析

定义类；构造函数重载；==运算符的重载。

2.程序源码

#include<iostream>

using namespace std;

class cuboid

{

private:

double a, b, c;

public:

cuboid()

{

a = 1; b = 1; c = 1;

}

cuboid(double, double, double);

void init(double, double, double);

double volumn() { return a\*b\*c; }

double area() { return 2 \* (a\*b + b\*c + c\*a); }

cuboid operator +(cuboid&ad)

{

cuboid temp;

temp.a = a + ad.a;

temp.b = b + ad.b;

temp.c = c + ad.c;

return temp;

}

bool operator ==(cuboid&ad)

{

if (ad.a == a&&ad.b == b&&ad.c == c)return true;

else return false;

}

~cuboid() { }

};

cuboid::cuboid(double aa, double bb, double cc)

{

a = aa; b = bb; c = cc;

}

void cuboid::init(double aa, double bb, double cc)

{

a = aa; b = bb; c = cc;

}

int main()

{

cuboid a(2.0, 3.0, 4.0);

cuboid b(3.0, 2.0, 1.0);

cuboid c(5.0, 5.0, 5.0);

cout << "b的体积：" << b.volumn() << endl;

cout << "b的表面积:" << b.area() << endl;

cuboid d;

d = a + b;

cout << "d的体积：" << d.volumn() << endl;

if (d == c)

{

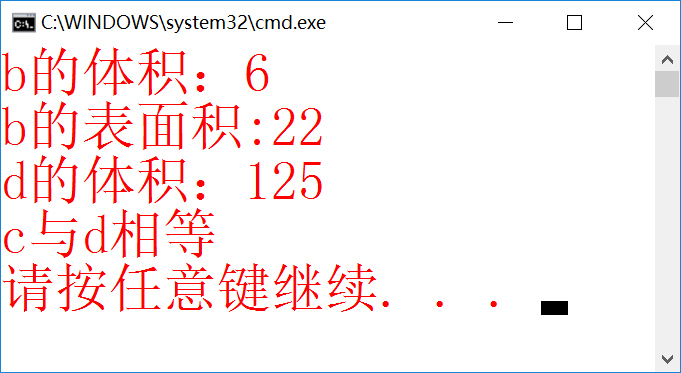
cout << "c与d相等" << endl;

}

return 0;

}

3.实验结果



题目三 定义一个空心圆形球体类，要求该类包含构造函数、能计算球体容积函数、总体积函数、外面积函数和内面积函数，运算符==重载函数。编写主函数加以测试。

1.要点分析

类的定义；运算符的重载。

2.程序源码

#include<iostream>

using namespace std;

#define PI 3.1415926

class ball

{

private:

double r, R;

double x, y, z;

public:

ball() { r = 1; R = 1; x = 1; y = 1; z = 1; }

ball(double rr, double RR, double xx, double yy, double zz)

{

r = rr; R = RR; x = xx; y = yy; z = zz;

}

void init(double rr, double RR, double xx, double yy, double zz)

{

r = rr; R = RR; x = xx; y = yy; z = zz;

}

double v1()

{

return PI\*r\*r;

}

double v2()

{

return PI\*R\*R;

}

double a1()

{

return 4 \* PI\*R;

}

double a2()

{

return 4 \* PI\*r;

}

bool operator==(ball &m)

{

if (r == m.r&&R == m.R)return true;

else return false;

}

};

int main()

{

double r, R, x, y, z;

cout << "请输入空心球的参数：" << endl;

cin >> r >> R >> x >> y >> z;

ball a(10.0, 20.0, 0.0, 0.0, 0.0);

ball b(r,R,x,y,z);

cout << "b球容积：" << b.v1() << endl;

cout << "b球总体积：" << b.v2() << endl;

cout << "b球外表面积:" << b.a1() << endl;

cout << "b球内表面积：" << b.a2() << endl;

if (b == a)

{

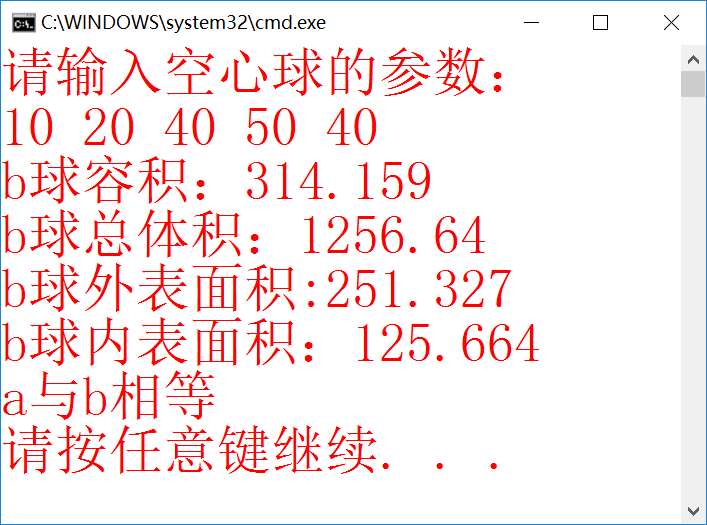
cout << "a与b相等" << endl;

}

return 0;

}

3.实验结果



题目四 设计定义正方形边框正小数类，类中包含三个数据成员：正方形整数边长、整数指针、浮点二维数组zfx[100][100]。类中至少包含无参和有参构造函数、生成正小数存储函数f\_zfx()、重载运算符==函数、输出正方形边框函数、析构函数。无参构造函数功能：边长设置为0，整数指针设置为NULL。有参构造函数功能：函数只有一个参数int n（即为整数边长），根据n，将类中成员整数指针指向动态申请一维整数数组，该数组长度为4\*(n-1)，随机生成4\*(n-1)个整数存入该数组。生成正小数存储函数f\_zfx()的功能：先将整数指针指向的一维数组中的整数转换为小数，然后以zfx[0][0]为起点，按顺时针方向将所有小数形成正方形边框存入浮点二维数组中，小数生成公式：某个整数+0.01\*i+0.0001\*j，i为行下标值，j为列下标值。输出正方形边框函数的功能：将浮点二维数组中的正方形边框的小数显示在屏幕上，即屏幕上基本上由小数构成正方形。重载运算符==指边长相等，且整数指针指向的一维数组元素对应相等，则两对象才相等。

1.要点分析

成员：存储边长的整型变量；整型指针；双精度数组的定义。

构造函数；运算符重载。

2.程序源码

#include <iostream>

#include<cmath>

using namespace std;

int i; //声明全局变量

class square //声明正方形框类

{

private:

int length; //边长

int \*p; //整型指针

double zfx[100][100]; //数组

public:

square() //无参构造函数

{

length = 0;

p = NULL;

for (i = 0; i < 100; ++i)

for (int j = 0; j < 100; ++j)

zfx[i][j] = 0;

}

square(int n) //有参构造函数

{

length = n;

p = new int[4 \* (n - 1)];

for (i = 0; i < 100; ++i)

for (int j = 0; j < 100; ++j)

zfx[i][j] = 0;

for (i = 0; i < 4 \* n - 4; ++i)

\*(p + i) = rand();

}

void f\_zfx() //初始化数组函数

{

int \*a = p;

for (i = 0; i < 4 \* (length - 1); ++i)

{

if (i < length - 1)

zfx[0][i] = \*(a + i) + 0.0001\*i;

else if (i >= length - 1 && i < length \* 2 - 2)

zfx[i % (length - 1)][length - 1] = \*(a + i) + 0.01\*(i % (length - 1)) + 0.0001\*(length - 1);

else if (i >= length \* 2 - 2 && i < length \* 3 - 3)

zfx[length - 1][length - 1 - i % (length - 1)] = \*(a + i) + 0.01\*(length - 1) + 0.0001\*(length - 1 - i % (length - 1));

else

zfx[length - 1 - i % (length - 1)][0] = \*(a + i) + 0.01\*(length - 1 - i % (length - 1));

}

}

void print() //输出函数

{

for (i = 0; i < length; ++i)

{

for (int j = 0; j < length; ++j)

{

if (zfx[i][j] == 0)

cout << " ";

else

{

cout.precision(10);

cout.width(15);

cout << zfx[i][j];

}

}

cout << endl;

}

cout << endl;

}

bool operator == (square ss) //重载==函数

{

if (length != ss.length)

return false;

else

{

int j = 1;

for (i = 0; i < 4 \* length - 1; ++i)

if (p[i] != ss.p[i])

{

j = 0;

break;

}

if (j == 1)

return true;

else return false;

}

}

~square() {}; //析构函数

};

int main() //声明主函数

{

square s1(5);

cout << "第一个正方形边框" << endl;

s1.f\_zfx();

s1.print();

square s2(5);

cout << "第二个正方形边框" << endl;

s2.f\_zfx();

s2.print();

square s3(4);

cout << "第三个正方形边框" << endl;

s3.f\_zfx();

s3.print();

if (s1 == s2)

cout << "第一第二个正方形边框相等" << endl;

else

cout << "第一第二个正方形边框不相等" << endl;

if (s1 == s3)

cout << "第一第三个正方形边框相等" << endl;

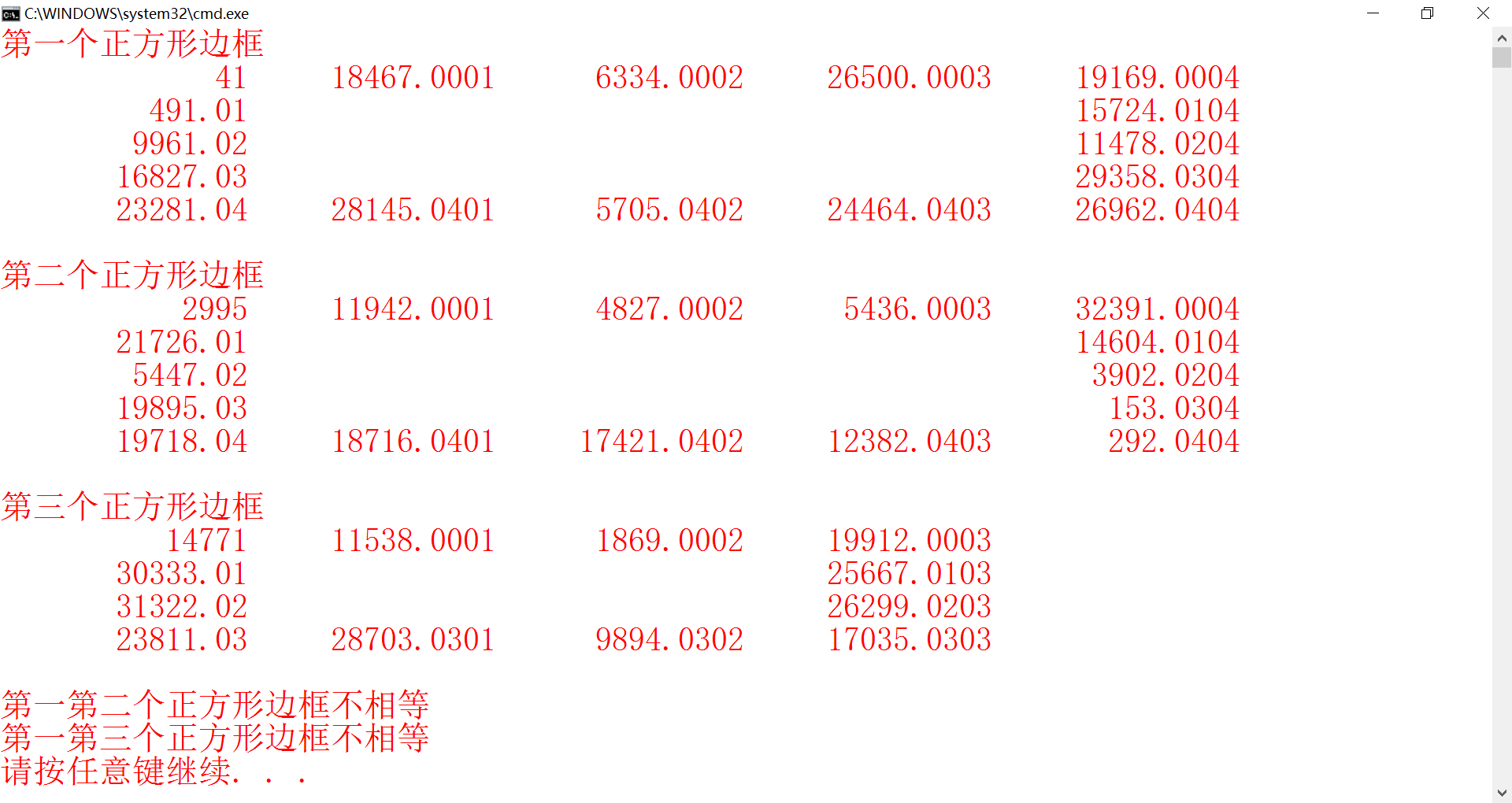
else

cout << "第一第三个正方形边框不相等" << endl;

return 0;

}

3.实验结果



题目五 从文件 mytest.txt （自己编写生成这个文本文件）读出全部内容，统计其中的单词 ( 以空格或’\n’为单词分隔符 ) 个数，并打印单词个数及总文件长度。

1.要点分析

输入输出；单词数判断。

2.程序源码

#include<iostream>

#include<fstream>

using namespace std;

int main()

{

char ch = '\0', ch1 = '\0';

int n = 1, m = -1, mm = 0;

ifstream in("mytest.txt");

if (!in) { cout << "Cannot open file!"; return 1; }

while (in)

{

m++;

if (ch == '\n') mm++;

in.get(ch);

cout << ch1;

if ((ch1 == '\n' || ch1 == ' ') && (ch != '\n'&&ch != ' ')) { n++; }

ch1 = ch;

}

cout << endl << "单词总数是" << n << endl;

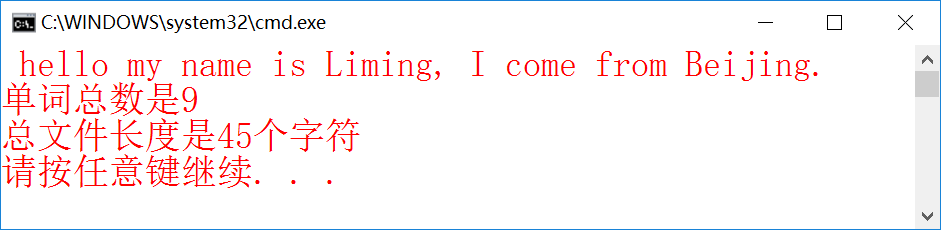
cout << "总文件长度是" << m - mm << "个字符" << endl;

in.close();

return 0;

}

3.实验结果



题目六 编写一个程序，可以读入一个数据文件 s\_data 中以 N × N 二维数组排列的数据（ N 不小于 3 ），将二维数组置逆，然后写入另一个文件 t\_data 中。   
假设 s\_data 文件中数据如下：   
45 87 32 34   
54 56 76 78   
46 78 87 23   
12 32 44 56   
则 t\_data 文件中的数据如下：   
45 54 46 12   
87 56 78 32   
32 76 87 44   
34 78 23 56

1.要点分析

输入输出；矩阵转置的算法。

2.程序源码

#include <iostream>

#include <fstream>

#include <cstring>

#include <cmath>

using namespace std;

int main()

{

int i = 0, j = 0;

double temp[100];

char name[20], c, number[10];

cout << "Please input file name:";

cin >> name;

ifstream in("s\_data.txt");

ofstream out(name);

if (!in && !out)

{

cout << "文件不能打开" << endl;

return 1;

}

while (in.get(c))

{

if (int(c) != 32)

{

number[i] = c;

++i;

}

else

{

number[i] = '\0';

i = 0;

temp[j] = atof(number);

++j;

}

}

int k = sqrt(j);

double \*\*a = new double \*[k];

for (i = 0; i < k; ++i)

a[i] = new double[k];

for (i = 0; i < k; ++i)

for (j = 0; j < k; ++j)

a[i][j] = temp[4 \* i + j];

for (i = 0; i < k; ++i)

{

for (j = 0; j < k; ++j)

out << a[j][i] << "\t";

out << endl;

}

for (i = 0; i < k; ++i)

delete[]a[i];

delete[]a;

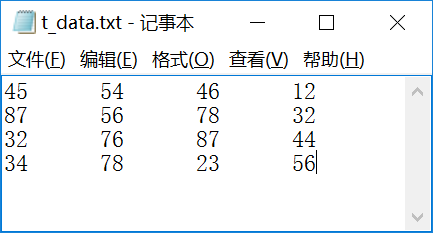
in.close();

out.close();

return 0;

}

3.实验结果



题目七 编写一个程序，可以读入一个文本文件（假设为data.txt），统计文件中大写字母个数，小写字母个数，数字字符个数，其它字符个数。并求出最长行（即该行中字符个数最多）的行号，输出所有结果。如果data.txt文件内容如下：  
假设data.txt文件中的数据如下：  
// Exercise 13\_5   
#include <iostream.h>   
return 0;   
}   
则输出：大写字母1个，小写字母29个，数字字符4，其它字符16。

1.要点分析

输入输出；字母和数字的统计。

2.程序源码

#include<iostream>

#include<fstream>

#include<cmath>

using namespace std;

int main()

{

int big = 0, small = 0, num = 0, line = 0, other = 0, word[100], max;

int i = 0, j;

char ch;

char name[20];

for (i = 0; i<100; i++)word[i] = 0;

cout << "Please input the file name:";

cin >> name;

ifstream in(name);

if (!in) { cout << "Cannot open the file!" << endl; return 0; }

while (in)

{

in.get(ch);

cout << ch;

word[line]++;

if (ch == '\n') line++;

if (ch>96 && ch<123) small++;

else if (ch>64 && ch<91) big++;

else if (ch>47 && ch<58) num++;

else { other++; }

}

in.close();

cout << endl << endl;

for (i = 0; i<line; i++) for (j = line - 1; j>i; j--) if (word[i]>word[j]) { max = i; }//找出最长行

cout << "最长行：" << "第" << max + 1 << "行" << endl;

cout << "大写字母" << big << "个" << endl;

cout << "小写字母" << small << "个" << endl;

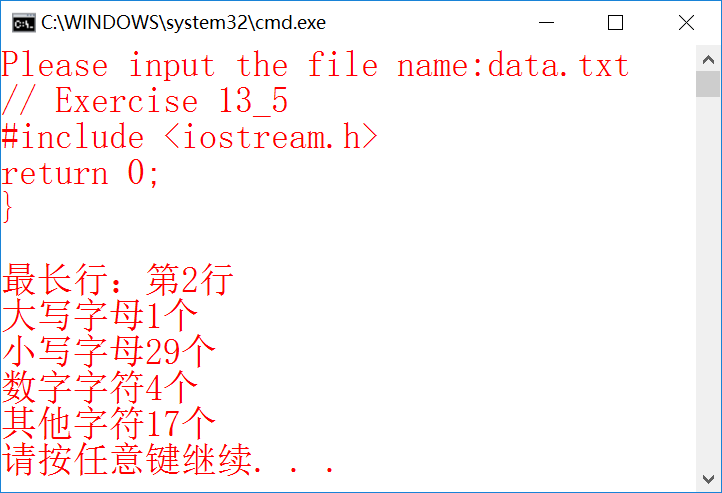
cout << "数字字符" << num << "个" << endl;

cout << "其他字符" << other - 1 << "个" << endl;

return 0;

}

3.实验结果



三、个人小结

上次就说是最后一次的个人小结，肺腑之言都掏了，情都抒得七七八八，结果这次又来了一次。那就说点实际的吧。

C++课结束了，马上钱班分流，我个人填了计算机方向。所以说，C++是逃不掉的。我早就有此觉悟，但奇怪的是学习过程中兴致一直不高。是上大学惰怠了，没有求知欲了？有一方面放松的原因，不过也不尽然。对这学期的状态说什么现在都为时已晚，但可以为以后的学习提供一点借鉴的经验。总的来说，练习才是进步的第一原动力。

同时阅读了上周杨海谦同学写的优秀实验报告。其优势主要体现在实验结果将各种讨论情况充分反馈，展现方式也较为多样；实验要点分析较为详尽，将题目结构拆解科学，没有矫情的凑字数步骤。