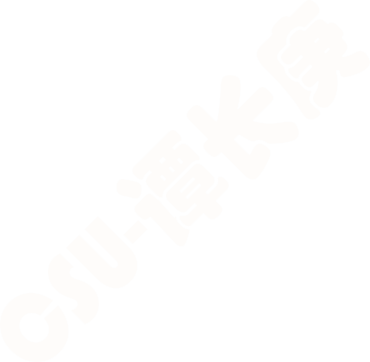
**计算机程序设计基础（C++）**

**实验报告**

专业班级 软工2206

学 号 8209220606

姓 名 杨庆豪

实验报告成绩：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 实验 | 实验二 | 实验三 | 实验四 | 实验五 | 总评 |
| 成绩 |  |  |  |  |  |

批阅教师：

**实验一 实验环境与简单程序设计**

1. **实验目的与要求**
2. 掌握集成开发环境，掌握C++程序的基本要素以及完整的C++程序开发过程。
3. 掌握基本数据类型、运算符和表达式的使用。理解隐式转换和强制转换，理解数据超过该数据类型标识范围时的溢出。掌握不同数据之间的混合算术运算中数据类型的转换。
4. 变量的定义与常量的使用。
5. 输入、输出的实现。
6. 编译信息的理解与错误的修改。
7. 简单程序的设计。
8. **实验内容**

项目1：编辑输入下列程序，找出下面代码的错误并改正：

include<iostream>

using namespace std;

int Main()

{

int i=k+1;

cout<<i++<< endl;

int i=1;

cout<<i++<< endl;

cout<<”Welcome to C++!<<endl;

return 0

}

仔细观察屏幕下方的信息框中编译器与连接器所给出的错误信息，了解其含义及改正方法。

项目2：求圆锥的体积：要求键盘输入圆锥底的半径、锥高，使用标识符常量定义圆周率。

1. 创建一个控制台项目
2. 在文件中输入程序内容，存盘
3. 编译、连接、运行；观察结果

项目3：通过下面程序验证你所使用系统上运行的C++编译器中每个基本数据类型的长度。

#include <iostream>

using namespace std;

int main()

{

cout << "char length:" << sizeof( char ) << endl;

cout << "int length:" << sizeof( int ) << endl;

return 0;

}

修改程序，验证short，long，float，double，long double，wchar\_t的类型长度。

项目4：观察下面程序的执行结果。

#include <iostream>

#include <iomanip>

using namespace std;

int main()

{

unsigned int testUnint=65534;//0xfffe

cout << "output in unsigned int type:" << testUnint<< endl;//<<oct;

cout << "output in char type:" << static\_cast<char>(testUnint)<< endl;

cout << "output in short type:" << static\_cast<short>(testUnint)<< endl; //为什么结果为-2？

cout << "output in int type:" << static\_cast<int> (testUnint)<< endl;

cout << "output in double type:"<< static\_cast<double>(testUnint)<< endl;

cout << "output in double type:" <<setprecision(4)<< static\_cast<double>(testUnint)<< endl;

cout << "output in Hex unsigned int type:" <<hex<< testUnint<< endl; //16进制输出 system("pause");

return 0;

}

自己编程测试一下将testUnint按8进制输出<<oct；将一个实数转换为int，观察结果。

项目5：编程，输入华氏温度，将其转换为摄氏温度后输出（保留两位小数）。

1. **实验步骤、算法与结果分析**

**项目1**

程序

using namespace std;

#include <iostream>

int main()

{

int k=1;

int i = k + 1;

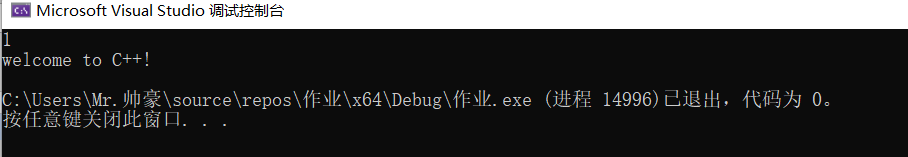
i = 1;

cout << i++ << endl;

cout << "welcome to C++!" << endl;

return 0;

}



最开始的main和ik的赋值都挺简单的，其中变量不能被重复定义是我忽视的，这一部分处理了比较久。

**项目2**、

using namespace std;

#include <iostream>

float yztj(float x, float y)

{

const float pai = 3.14;

float z;

z = 1.0 / (3.0)\* x \* x \* pai \* y;

return z;

}

int main()

{

float x, y;

cin >> x >> y;

cout << yztj(x, y) << endl;

return yztj(x, y);



一个比较基础的函数定义和调用，只是除法中int和float的区别要注意，不然结果就一直错

**项目3**.

return 0;

}

修改程序，验证 short,long,float,double,long double,wchar\_t 的类型长度。

#include<iostream>

using namespace std;

int main()

{

cout << "short length" << sizeof(short)<<"\0"<<endl;

cout << "long length" << sizeof(long)<<"\0" <<endl ;

cout << "float length" << sizeof(float) << "\0"<<endl;

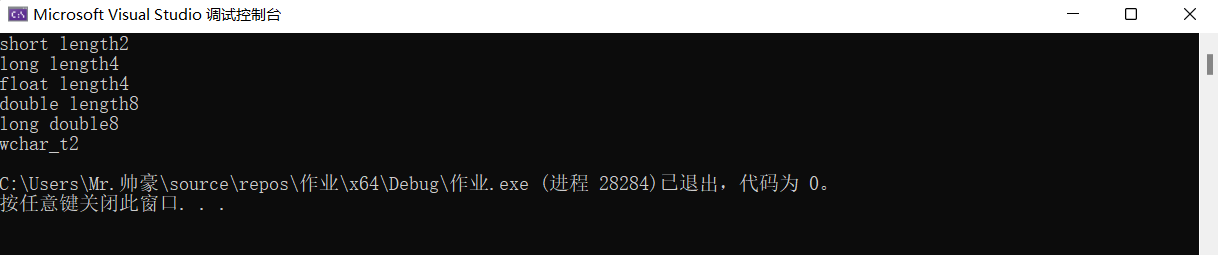
cout << "double length" << sizeof(double)<<"\0"<<endl;

cout << "long double" << sizeof(long double)<<"\0" <<endl;

cout << "wchar\_t" << sizeof(wchar\_t)<<"\0" <<endl ;

return 0;

}



很正常的sizeof取类型数据长度。

**项目4**、

#include <iostream>

#include <iomanip>

using namespace std;

int main()

{

unsigned int testUnint = 65534;

cout << "output in unsigned int type:" << testUnint << endl;

cout << "output in short type:" << static\_cast<short>(testUnint) << endl;

cout << "output in double type:" << static\_cast<double>(testUnint) << endl;

cout << "output in double type:" << setprecision(4) << static\_cast<double>(testUnint) << endl;

cout << "output in Hex unsigned int type:" << hex << testUnint << endl;

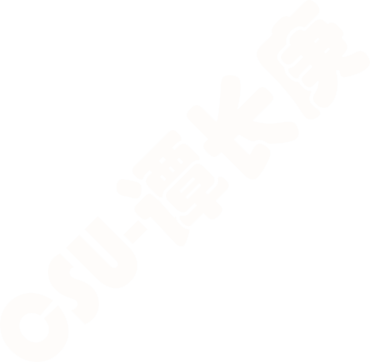
cout << "output in Hex unsigned int type:" << oct<< testUnint << endl;

system("pause");

return 0;

}



**项目****5**、

#include<iostream>

using namespace std;

float wd(float a)

{

float b = 32 + 1.8 \* a;

cout << b << endl;

return b;

}

int main()

{

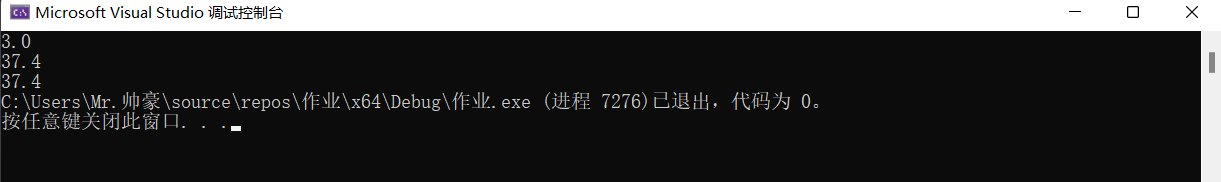
float x;

cin >> x;

cout << wd(x);

return 0;

}



1. 遇到的问题和解决方法

总体来说，实验一的内容感觉都不是很难，但在一些细节问题却出了问题，比如项目一的变量无法被重复定义，项目二中除法中数据浮点数和整数型的区别，项目四中输出八进制的知识盲区，前两个问题在编译器和自己反复的分析下翻书解决了，第三个问题在上网查阅了相关的资料后学会了关于转换成八进制的新的知识。

体会

实验一让我感受到c++这门课的实验属性，虽然书本的基础知识肯定是必不可少的，但在做完实验一后我感觉到，缺乏了实验的实践经验，书本上的知识不过是空中楼阁，并且会存在许多的瑕疵和纰漏。比如说变量不能重复定义和除法对不同数据类型作用效果不一样的知识，虽然我是清楚的，但真的上机实操后才真正意识到这个问题。因此，以后一定要多多实践，多多打代码。

**项目**1、

**程序：**

#include <iostream>

using namespace std;

int main()

{

char a;

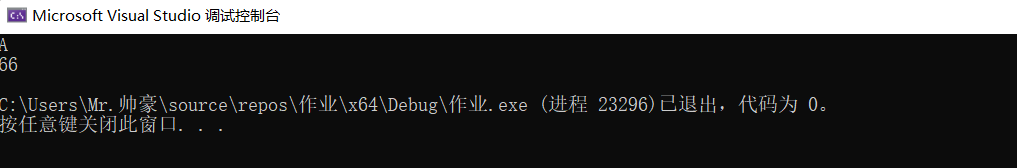
cin>>a;

if (a <= 'Z' && a >= 'A')

cout <<a+1 << endl;

else cout<<char(a-32)<<endl;

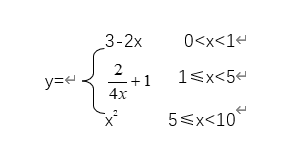
}





**项目**2、

**程序：**



#include<iostream>

using namespace std;

int main()

{

float x;

cin >> x;

if (x > 0 && x < 10)

{

if (x >= 5)

{

cout << x \* x << endl;

}

else if (x >= 1)

{

cout << 2 / 4 \* x + 1 << endl;

}

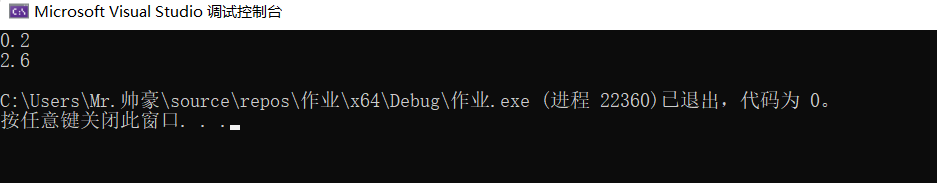
else cout << 3 - 2 \* x << endl;

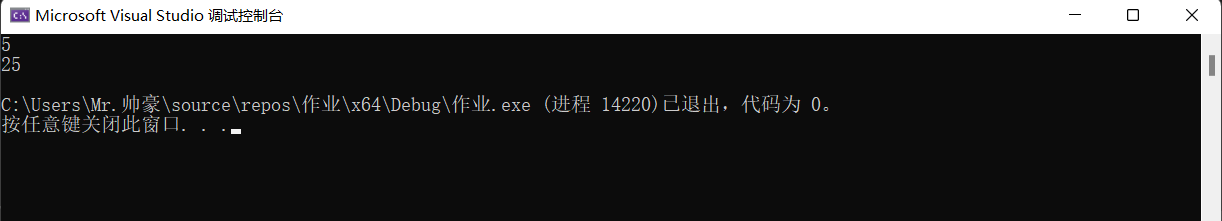
}

else cout << "不属于定义域" << endl;

return 0;

}









**项目3**

#include<iostream>

using namespace std;

float zc(float x, float y, float z)

{

float C = x + y + z;

return C;

}

int main()

{

float a, b, c;

cin >> a>>b>>c;

cout << zc(a, b, c) << "\t" << endl;

if (a + b > c && a +c < b&&b+c>a)

{

if (a == b || a == c || b == c)

cout << "this is a deng yao san jiao xing";

else cout << "zhe bu shi deng yao san jiao xing" << endl;

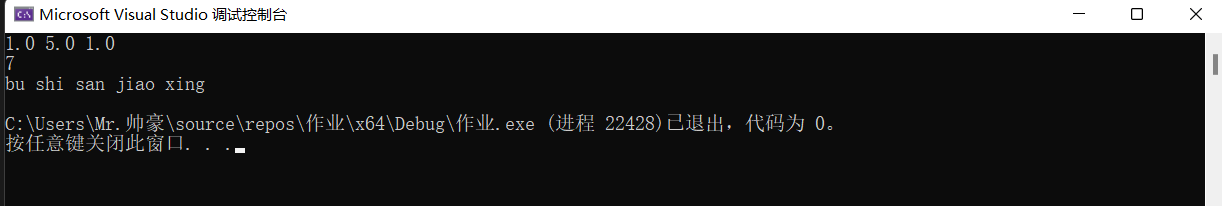
}

else cout << "bu shi san jiao xing" << endl;

return 0;

}





**项目**4、

**程序:**

#include<iostream>

using namespace std;

int main()

{

int a = 0, b = 0, c = 0, d = 0;

char x;

cout << "请输入一串任意字符： \n";

while ((x = getchar()) != '\n')

{

if (x >= 'a' && x <= 'z' || x >= 'A' && x <= 'Z')

{

a++;

}

else if (x == ' ')

{

b++;

}

else if (x >= '0' && x <= '9')

{

c++;

}

else

{

d++;

}

}

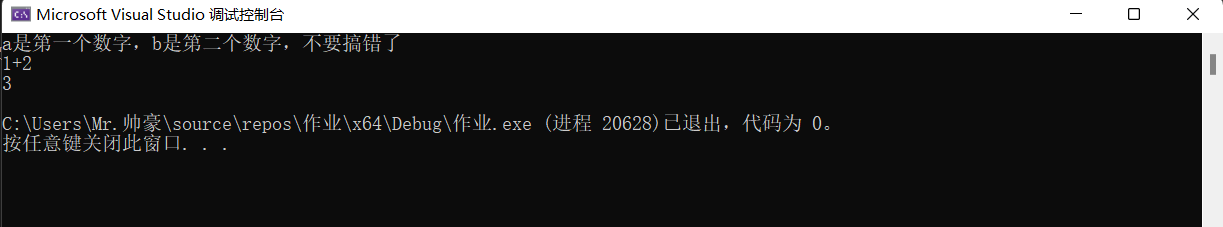
cout << "letter :" << a << endl;

cout << "space :" << b << endl;

cout << "number :" << c << endl;

cout << "other :" << d << endl;

}



**项目5**

**程序:**

#include<iostream>

using namespace std;

int main()

{

int a = 0, b = 0, c = 0, d = 0;

char x;

cout << "请输入一串任意字符： \n";

while ((x = getchar()) != '\n')

{：

if (x >= 'a' && x <= 'z' || x >= 'A' && x <= 'Z')

{

a++;

}

else if (x == ' ')

{

b++;

}

else if (x >= '0' && x <= '9')

{

c++;

}

else

{

d++;

}

}

cout << "letter :" << a << endl;

cout << "space :" << b << endl;

cout << "number :" << c << endl;

cout << "other :" << d << endl;

}

****

**项目**6

#include<iostream>

using namespace std;

int main()

{

int a, b, c, d, e;

cin >> a >> b;

if (b > a)

{

d = a;

a = b;

b = d;

}

e = a \* b;

do {

c = a % b;

if (c == 0)

{

cout << "最大公约数为：" << b << endl;

break;

}

a = b;

b = c;

} while (c != 0);

e = e / b;

cout << "最小公倍数为：" << e << endl;

return 0;



**项目7**

**程序：**

#include<iostream>

using namespace std;

int main()

{

int a, b, i;

for (i = 1; i <= 5; i++)

{

for (a = 5; a >= i - 1; a--)

{

cout << " ";

}

for (b = 1; b <= i; b++)

{

cout << "\*";

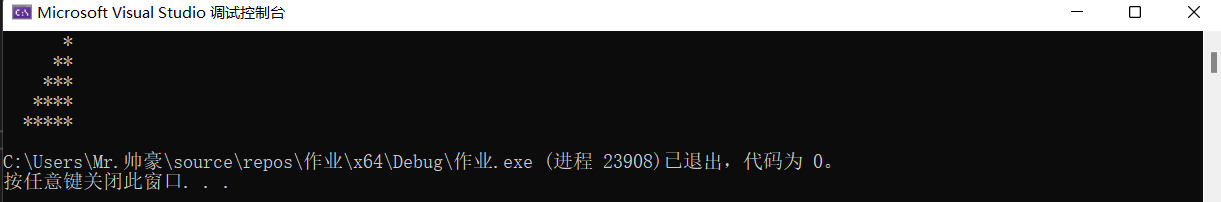
}

cout << '\n';

}

return 0;

}



**项目8**

**程序：**

#include<iostream>

#include<iomanip>

using namespace std;

int main()

{

double a, b, c;

cout << "please input a number:";

cin >> a;

b = a;

c = (b + a / b) / 2;

do

{

b = c;

c = (b + a / b) / 2;

} while (c - b < -0.00001 || c - b > 0.00001);

cout << "平方根为" << setprecision(5) << c << endl;

return 0;

}

****

**改进**

#include<iostream>

#include<iomanip>

using namespace std;

int main()

{

double a, b, c;

cout << "please input a number:";

cin >> a;

b = a;

c = (b + a / b) / 2;

if (a < 0)

{

cout << "error" << endl;

}

do

{

b = c;

c = (b + a / b) / 2;

} while (c - b < -0.00001 || c - b > 0.00001);

cout << "平方根为" << setprecision(5) << c << endl;

return 0；}

****

**思考2**

#include<iostream>

#include<iomanip>

using namespace std;

int main()

{

double a, b, c;

cout << "please input a number:";

cin >> a;

b = a;

c = (b + a / b) / 2;

if (a < 0)

{

cout << "error" << endl;

}

do

{

b = c;

c = (b + a / b) / 2;

} while (c - b > -0.000000000001 || c - b < 0.0000000000001);

cout << "平方根为" << setprecision(11) << c << endl;

return 0;

}

****

**结果，六位小数后认为两个数据相同，无法实现**

**项目9**

**程序：**

#include<iostream>

using namespace std;

int main()

{

float m = 0, i, j;

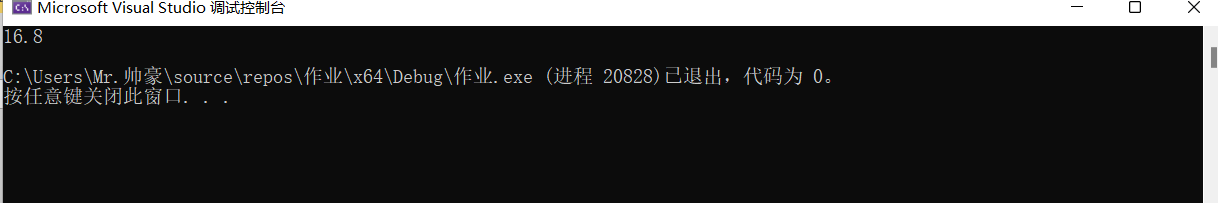
for (i = 0, j = 2; j <= 100; j=j \* 2)

i++, m = m + j;

cout << m / i \*0.8<< endl;

return 0;

}

****

1. **遇到的问题和解决方法**

**实验二明显比实验一要难很多，其中最困惑我的是项目六，因为一开始我想通过动态数组的方式存储输入的自负串，结果发现很麻烦，上网查阅相关方法之后才知道了getchar（）这个新的函数，学会了新的知识。在编写星星树的时候，我完全想不出怎么让星星靠左排放，也是在上网查询和询问同学后我才知道用嵌套循环来实现。**

1. **体会**

**在编写实验二中的程序时，发现死记硬背书上的知识是没有意义的，必须要灵活运用，比如困惑我许久的星星树就是其中典型的案列。不仅如此，在做实验的过程中，我发现，学习不能拘泥于课本，还要课外扩展，利用线上线下的各种资源，如csdn来学习各种各样的新知识。如getchar（）函数的运用，以及最新的知识，如strcpy\_s这个安全性能更高的函数。**