**计算机程序设计基础（C++)**

**实验报告**

专业班级： 软工2401

学 号： 8209240104

姓 名： 丛非凡

**实验报告成绩：**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **实验** | **实验一** | **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**

**一、实验目的**

1、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。

2、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

表示范围时的溢出。掌握不同数据之间的混合算术运算中数据类型的转换。

3、变量的定义与常量的使用。

4、输入、输出的实现。

5、编译信息的理解与错误的修改。

6、简单程序的设计。

**二、实验内容**

熟悉C++编程环境，可以使用VS；对已经能熟练掌握C++开发环境的同学，可

以跳过本部分内容）

**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;

}

**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

**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;

}

**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;//oxfffe

cout << "output in unsigned int 1 type:"" << testUnint<< end;//<<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;je\_将一个实数转换成int,观察结果。**

**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）。**

**三、算法分析，程序结果**

1

第一个int的I是大写的，k未定义，同一作用域内不能重复定义同名变量。

改正后：

#include<iostream>

using namespace std;

int main()

{

int k = 0;

int i = k + 1;

cout << i++ << endl;

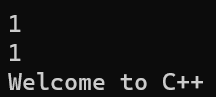
i = 1;

cout << i++ << endl;

cout << "Welcome to C++" << endl;

return 0;

}



2.

Pi为常量3.14，圆锥体积公式PI\*r^2\*h/3，注意输入输出即可。

#include <iostream>

using namespace std;

#define PI 3.14

int main()

{

float r, h;

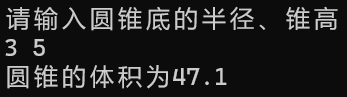
cout << "请输入圆锥底的半径、锥高" << endl;

cin >> r >> h;

float v = PI \* r \* r \* h / 3.0f;

cout << "圆锥的体积为" << v<<endl;

return 0;

}

3.

#include<iostream>

using namespace std;

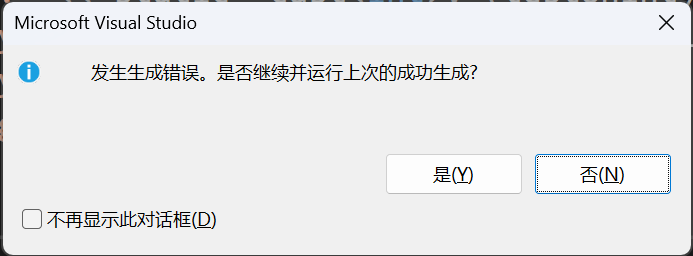
int main()

{

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

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

}

4.

源代码出现错误，改正并用八进制输出代码如下

#include <iostream>

#include <iomanip>

using namespace std;

int main() {

unsigned int testUnint = 65534; //oxfffe

cout << "output in unsigned int 1 type:" << testUnint << endl;

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?: unsigned int 变量强制转换为 short 类型(取值范围是 -32,768 到 32,767)时，发生了溢出，65534 经过溢出计算后，变成了 65534 - 65536 = -2。

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

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

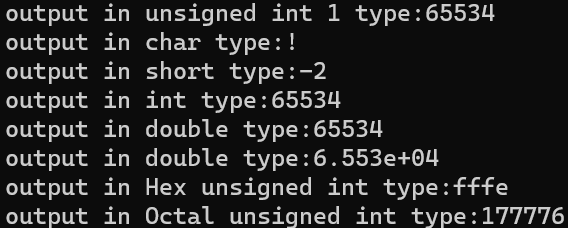
cout << "output in double type:" << setprecision(4) << static\_cast<double>(testUnint) << endl;

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

cout << "output in Octal unsigned int type:" << oct << testUnint << endl;//8进制输出

system("pause");

return 0;

}

为什么结果为-2?

unsigned int 变量强制转换为 short 类型(取值范围是 -32,768 到 32,767)时，发生了溢出，65534 经过溢出计算后，变成了 65534 - 65536 = -2。

5.

摄氏度与华氏度之间的转换关系：C=(F-32)÷1.8，另外注意保留两位小数

#include<iostream>

#include<iomanip>

using namespace std;

float FtoC(float F)

{

float C;

C = (F - 32)/1.8;

return C;

}

int main()

{

float C, F;

cout << "请输入华氏度" << endl;

cin >> F;

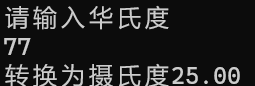
C= FtoC(F);

cout << fixed << setprecision(2);

cout << "转换为摄氏度" << C << endl;

return 0;

}



**四、遇到的问题与解决方法**

保留两位小数的方法不会，与同学交流后得出解决方案。

**五、体会**

编程需要多加练习不能眼高手低，还需要多多学习交流提升技术。

**实验二、数据结构**

**一、实验目的**

1、学习与掌握逻辑运算与逻辑表达式。

2、熟练掌握if、switch、while、do-while，for语句的语法结构与执行过程。

3、掌握选择、循环程序的设计方法

**二、实验内容**

1、输入一个字符，如果为小写，转换为大写输出，否则，输出其后继字符的ASCII码值。

2、输入x计算表达式的值：



分别输入 0.2, 1, 5 , 0,观察输出结果。

1. 输入三角形的三条边，求周长，并判断该三角形是否为等腰三角形（提示：要三边是否可以构成三角形）。

4、 完成计算器程序，实现（+ - \* / %）运算。考虑除数为 0 与运算符非法的情况。

5、输入一行字符，分别统计出其中英文字母、空格、数字字符和其它字符的个数。

提示：从键盘上读入一个字符给变量 c，判断 c 是属于哪种字符并计数，循环读入下个字符，直到回车换行字符'\n'为止。

cin，scanf（）都不能读入空格以及‘\n’字符，查找资料解决输入这两个字符的方法。

（这个题训练大家自主学习能力以及如何获取新知识、探索解决未知问题的能力。）

6、编写一个程序：从键盘上输入两个正整数，求 a 和 b 的最大公约数与最小公倍数。

7、使用循环结构输出下列图形：

\*

\*\*

\*\*\*

\*\*\*\*

\*\*\*\*\*

8、从键盘输入 a，用迭代法求 a 的平方根 x= *a* 。求平方根的迭代公式为：



要求精确到|xn+1 - xn|<10 -5。

提示：迭代法是把 xn代入迭代公式右边，计算出 xn+1来，然后把 xn+1 作为新的 xn ，计算出新的 xn+1，如此重复，直到|xn+1 - xn|<10 -5 时，xn+1 为所求的平方根。可以把 a 作为 xn 的初始值。

思考：（1）如果输入 a 为负，在运行时会出现什么情况? 修改程序使之能处理任何的 a 值。

(2）能否|xn+1 -xn|<10 -10或更小? 为什么? 请试一下。

9、苹果每个 0.8 元，第一天买 2 个，第二天开始，每天买前天的 2 倍，直到购买的苹果数不超过100的最大值，求每天平均花多少钱。

**三、算法分析，程序结果**

1

注意char与int的转换关系

#include <iostream>

using namespace std;

int main()

{

char p;

cin >> p;

if (p >= 'a' && p <= 'z')cout << char(p + 'A' - 'a') << endl;

else cout << p + 1 << endl;

return 0;

}

2

注意到x<=0或x>=10时为非法数据，输出-1表示非法数据

#include <iostream>

using namespace std;

float func(float x)

{

if (x <= 0||x>=10)return -1;

if (x < 1)return (3 - 2\*x);

if (x < 5)return (2.0 / (4.0 \* x) + 1);

if (x < 10)return x \* x;

}

int main()

{

float p;

while (cin >> p)cout << func(p) << endl;

return 0;

}

3

注意不能构成三角形的情况除了两边之和小于等于第三边，还要边长合法（大于0），虽然使用下面方法可以避免判断该情况。另外还需注意三元取最大值的方法。

#include <iostream>

using namespace std;

int main()

{

float a, b, c;

cout << "请输入三角形三边" << endl;

cin >> a >> b >> c;

if (a + b + c <= 2 \* max(max(a, b),c))cout << "不能构成三角形" << endl;

else

{

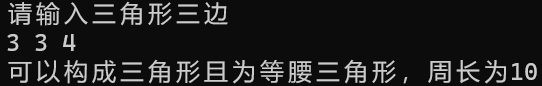
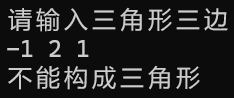
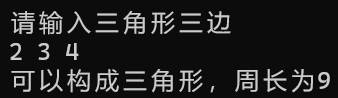
cout << "可以构成三角形";

if (a == b || b == c || a == c)cout << "且为等腰三角形";

cout << "，周长为" << a + b + c << endl;

}

return 0;

****}

4

由题意应当实现整数的四则运算与取模运算，则不需要注意%不能取浮点数的问题，只需要注意除数为0

#include <iostream>

using namespace std;

int main()

{

int a, b,ans;

char op;

cout << "请输入" << endl;

cin >> a >> op >> b;

switch (op)

{

case '+':

{

ans = a + b;

break;

}

case '-':

{

ans = a - b;

break;

}

case '\*':

{

ans = a \* b;

break;

}

case '/':

{

if (b == 0)

{

cout << "输入不合法" << endl;

return 0;

}

ans = a / b;

break;

}

case '%':

{

if (b == 0)

{

cout << "输入不合法" << endl;

return 0;

}

ans = a % b;

break;

}

default:

{

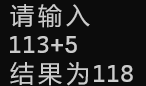
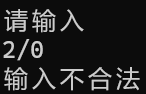
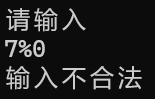
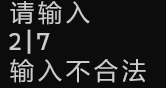
cout << "输入不合法" << endl;

return 0;

}

}

cout << "结果为" << ans;

}

5

使用getchar即可

#include <iostream>

using namespace std;

int mp[10] = { 0 };//1：中英文字母2：空格3：数字字符4：其它字符

int main()

{

char p;

while (p = getchar())

{

if (p == '\n')break;

if ((p >= 'a' && p <= 'z') || (p >= 'A' && p <= 'Z'))mp[1]++;

else if (p == ' ')mp[2]++;

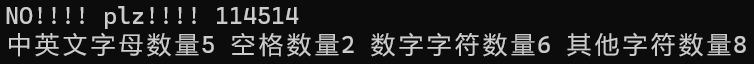
else if (p >= '0' && p <= '9')mp[3]++;

else mp[4]++;

}

cout << "中英文字母数量" << mp[1] << " 空格数量" << mp[2] << " 数字字符数量" << mp[3] << " 其他字符数量" << mp[4] << endl;

return 0;

}

6

使用欧几里得算法来求即可

#include <iostream>

using namespace std;

int gcd(int a, int b) { return b == 0 ? a : gcd(b, a % b); }

int lcm(int a, int b) { return a \* b / gcd(a, b); }

int main()

{

int a, b;

cin >> a >> b;

cout << "最大公约数为" << gcd(a, b) << "最小公倍数为" << lcm(a, b) << endl;

return 0;

}

7

简单的模拟

#include <iostream>

using namespace std;

int main()

{

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

{

for (int j = 1; j <= i; j++)cout << '\*';

cout << '\n';

}

return 0;

}

8

注意a为0或负数时单独拿出来

#include <iostream>

using namespace std;

int main()

{

double a, x,xp;

cin >> a;

if (a < 0)

{

cout << "输入不合法" << endl;

return 0;

}

if (a == 0)

{

cout << "其平方根为" << 0 << endl;

return 0;

}

xp = x= a;

do

{

xp = x;

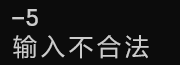
x = (xp + a / xp) / 2.0;

} while (abs(x - xp) >= 10e-5);

cout <<"其平方根为" << x << endl;

return 0;

}

1. 输出-inf或死循环
2. 浮点型表示精度有限，10e-10无意义

9

注意最后一天购买苹果后也不能超过100个

#include <iostream>

using namespace std;

#define price 0.8

int main()

{

double sum=2, d=2,money = 0,days = 0;

while (sum<=100)

{

days++;

money += d \* price;

d \*= 2;

sum += d;

}

cout << "每天平均花" << money / days << endl;

return 0;

}

**四、遇到的问题与解决方法**

Gcd的求法不会，询问班助解决。

**五、体会**

要多注意边界情况，防止非法输入