**C++实验报告一： 断点的使用**

**刘振宇 自18G2 201830312021**

**设计程序的思路：**

设计一个简单的输出“Hello，world！I am+你的年龄 ”的c++程序，使用cout 和cin 实现对键盘键入值的显示，输出标题“Please input your AGE”。提示输入你的年龄，当从键盘输入你的年龄，执行后会显示程序输出“Hello，world！I am+你的年龄 ”

**程序**：

#include<iostream>

using namespace std;

int main()

{

int age;

cout<<"Please input you AGE"<<endl;

cin>>age;

cout<<"hello,world! I am"<<age<<endl;

return 0;

}

**常见问题：**

程序运行秒退

**法一：**添加头文件

conio.h头文件

#include<iostream>

//#include<conio.h>

usingnamespacestd;

intmain()

{

intage;

cout<<"Please input you AGE"<<endl;

cin>>age;

cout<<"hello,world! I am "<<age<<endl;

//getch();

return0;

}

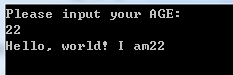
一开始输入的程序里是没有这两句被注释掉的语句的，结果程序运行会发生闪退的情况，无法正常运行显示输出内容。

之后跟随老师的讲解我把这两句程序添加进去，然后闪退的问题就解决了。

getch()这个函数是一个不回显函数，当用户按下某个字符时，函数自动读取，无需按回车

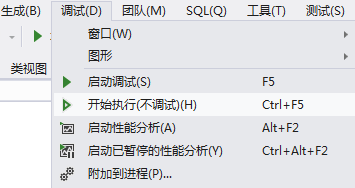
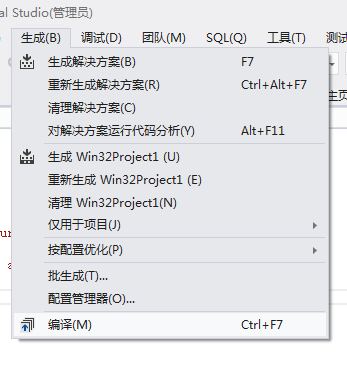
程序正常运行并显示输出：“Hello，world！I am+你的年龄 ”

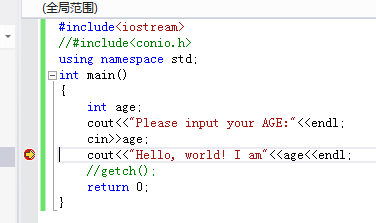
运行结果：



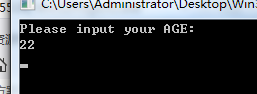
**法二：**

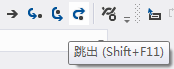
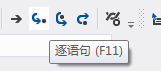
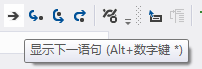
如果不用getch（）这个函数，也可以运用调试菜单里的开始执行（不调试），也可以达到同样的显示程序输出的效果。

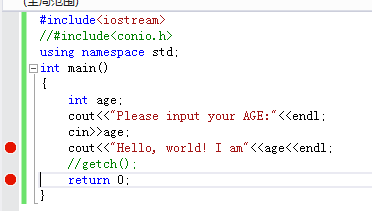




另外我还发现如果对程序添加一个或多个断点，那我就可以选择逐步执行一步一步的观察程序每个细节的运行情况，有助于我详细了解C++的程序思想和运行优先顺序。



断点运行跳到C++内部语句



**第一次作业：（见附件）**

显示自己学号的最后两位

程序：

#include<iostream>

#include<conio.h>

usingnamespacestd;

intmain()

{

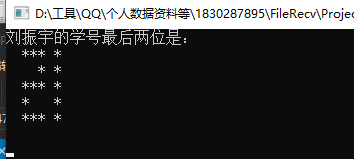
cout<<"刘振宇的学号最后两位是："<<endl;

cout<<" \*\*\* \*\n \* \*\n \*\*\* \*\n \* \*\n \*\*\* \*\n"<<endl;

getch();

return0;

}



问题：

1、单引号&双引号的区别：

单引号‘’字符

双引号“”字符串

2、生成和调试的区别：

生成：顺序执行程序，不检查程序语句错误

调试：可以开始执行（不调试）加断点逐步运行查看程序运行结果，可以利用C++内部程序检查程序语句