



§. 基础知识题 – 循环结构

要求:

- 1、完成本文档中所有的题目并写出分析、运行结果
- 2、无特殊说明，均使用VS2019编译即可
- 3、直接在本文件上作答，**写出答案/截图（不允许手写、手写拍照截图）**即可；填写答案时，为适应所填内容或贴图，**允许调整**页面的字体大小、颜色、文本框的位置等
 - ★ 贴图要有效部分即可，不需要全部内容
 - ★ 在保证一页一题的前提下，具体页面布局可以自行发挥，简单易读即可
 - ★ **不允许**手写在纸上，再拍照贴图
 - ★ **允许**在各种软件工具上完成（不含手写），再截图贴图
 - ★ 如果某题要求VS+Dev的，则如果两个编译器运行结果一致，贴VS的一张图即可，如果不一致，则两个图都要贴
- 4、转换为pdf后提交
- 5、**10月14日前**网上提交本次作业（在“文档作业”中提交）

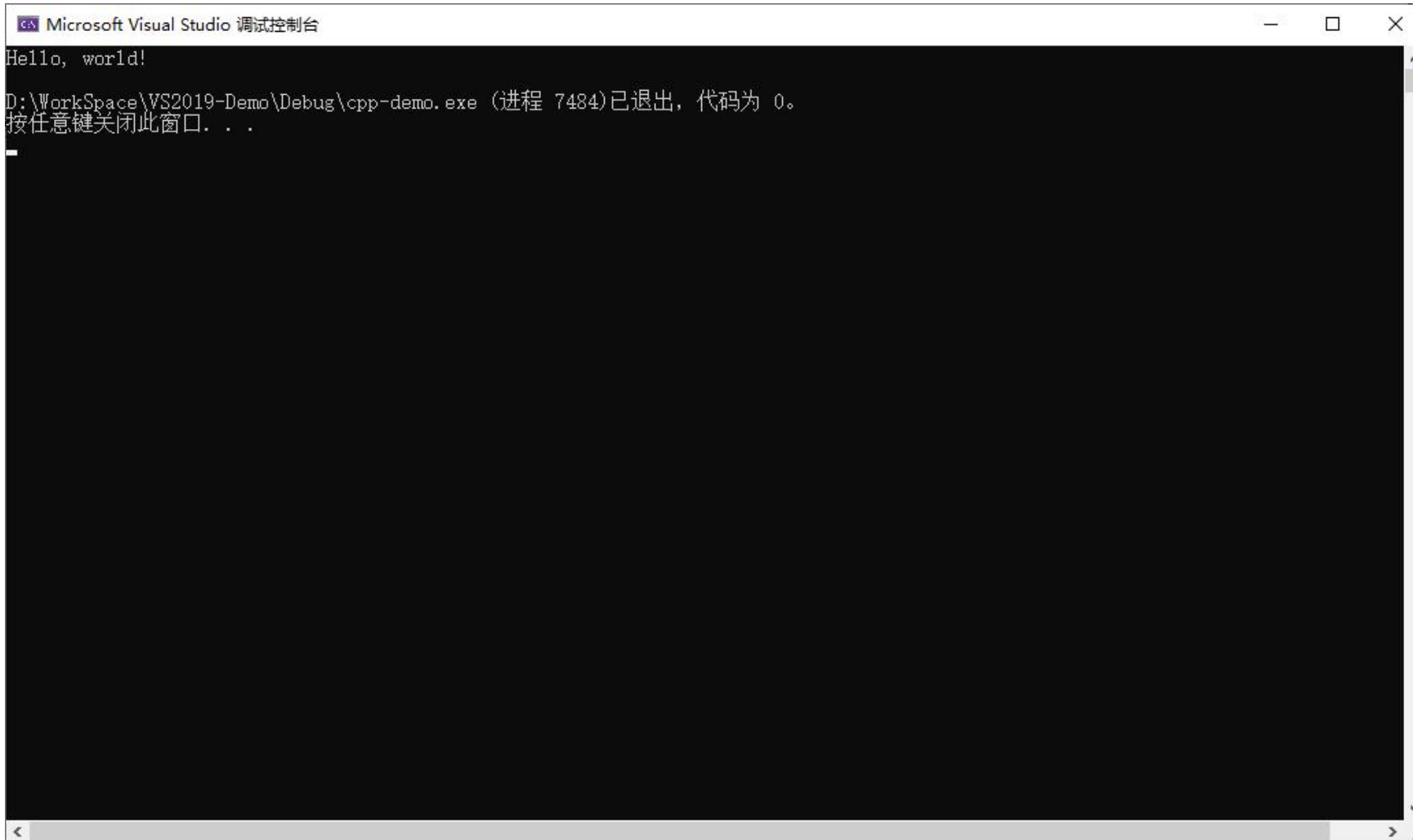
=> 注：因为课时问题，本次作业10069206/10071706班级的同学要10.12才结束课程，但因为题目数量较少，且本作业不影响本周编程作业（选择结构）的完成，因此截止日期仍保持10.14不变



§. 基础知识题 - 循环结构


贴图要求：只需要截取输出窗口中的有效部分即可，如果全部截取/截取过大，则视为无效贴图

例：无效贴图



```
Microsoft Visual Studio 调试控制台
Hello, world!
D:\Workspace\VS2019-Demo\Debug\cpp-demo.exe (进程 7484)已退出, 代码为 0.
按任意键关闭此窗口. . .
```

例：有效贴图



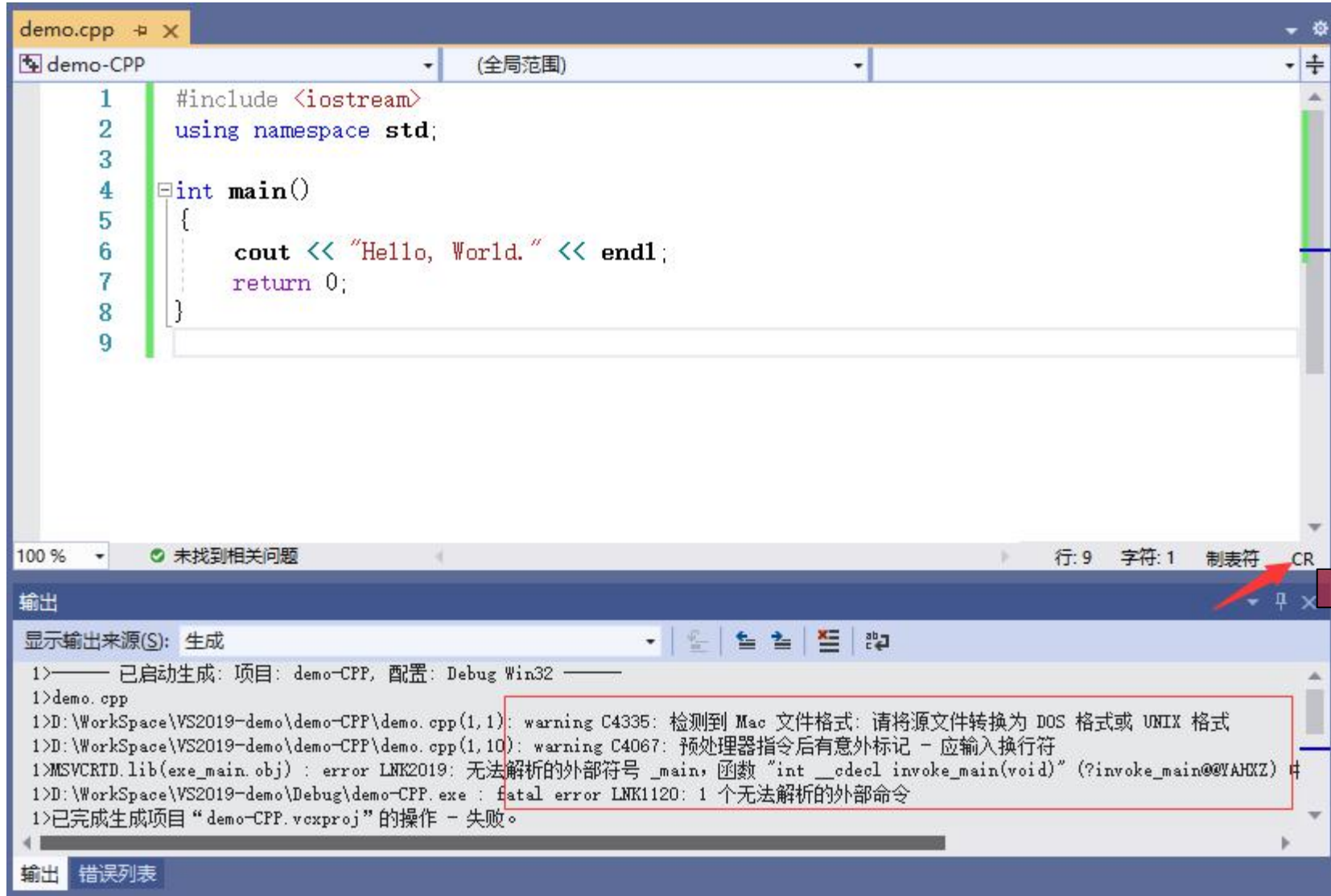
```
Microsoft Visual Studio 调试控制台
Hello, world!
```



§. 基础知识题 - 循环结构

附：用WPS等其他第三方软件打开PPT，将代码复制到VS2019中后，如果出现类似下面的**编译报错**，则观察源程序编辑窗

的右下角是否为CR，如果是，单击CR，在弹出中选择CRLF，再次CTRL+F5运行即可





§. 基础知识题 – 循环结构

1、循环的嵌套

A. 观察下列程序的运行结果，回答问题并将程序的运行结果截图贴上(如果有错则贴错误信息截图)

```
#include <iostream>
using namespace std;

int main()
{
    int i, j, k;
    int count1 = 0, count2 = 0, count3 = 0;

    for(i=1; i<=100; i++) {
        ++count1;
        for(j=1; j<=100; j++) {
            ++count2;
            for(k=1; k<=100; k++)
                ++count3;
        }
    }

    cout << "count1=" << count1 << endl;
    cout << "count2=" << count2 << endl;
    cout << "count3=" << count3 << endl;
    return 0;
}
```

1、贴运行结果

```
Microsoft Visual Studio 调试控制台
count1=100
count2=10000
count3=1000000
```

2、当循环嵌套时，内层循环的执行次数和外层循环是什么关系？

答：若每层循环的初值为1，则内层循环次数=内层一次for循环的次数*外层一次for循环的次数。



§. 基础知识题 – 循环结构

1、循环的嵌套

B. 观察下列程序的运行结果，回答问题并将程序的运行结果截图贴上(如果有错则贴错误信息截图)

```
#include <iostream>
using namespace std;

int main()
{
    int i, j, k;
    int count1 = 0, count2 = 0, count3 = 0;

    for(i=1; i<=100; i++) {
        ++count1;
        for(j=i; j<=100; j++) {
            ++count2;
            for(k=j; k<=100; k++)
                ++count3;
        }
    }

    cout << "count1=" << count1 << endl;
    cout << "count2=" << count2 << endl;
    cout << "count3=" << count3 << endl;
    return 0;
}
```

1、贴运行结果

```
Microsoft Visual Studio 调试控制台
count1=100
count2=5050
count3=171700
```

2、当循环嵌套时，内层循环的执行次数和外层循环是什么关系？

答：若每层循环的初值为外一层循环变量的值，而每个for循环次数为k次，当外层循环执行第n次，内层循环执行次数增加(k+1-n)次。

§. 基础知识题 - 循环结构



此页不要删除，也没有意义，仅仅为了分隔题目



§. 基础知识题 – 循环结构

2、break与continue

A. 已知代码如下，回答问题

```
while(1) {  
    ①  
    ②  
    if (X)  
        continue;  
    ③  
    ④  
}
```

当X为真时，重复执行①② (①②③④)

当X为假时，重复执行①②③④ (①②③④)

```
for(1; 1; ④) {  
    ①  
    ②  
    if (X)  
        continue;  
    ③  
}
```

当X为真时，重复执行①②④ (①②③④)

当X为假时，重复执行①②③④ (①②③④)



§. 基础知识题 – 循环结构

2、break与continue

B. 观察下列程序的运行结果，回答问题并将程序的运行结果截图贴上(如果有错则贴错误信息截图)

```
#include <iostream>
using namespace std;

int main()
{
    int i=0, sum=0;

    while(i<1000) {
        i++;
        break;
        sum=sum+i;
    }

    cout << "i=" << i << endl;
    cout << " sum=" << sum << endl;

    return 0;
}
```

Microsoft Visual Studio 调试控制台

```
i=1
sum=0
```

```
#include <iostream>
using namespace std;

int main()
{
    int i=0, sum=0;

    while(i<1000) {
        i++;
        continue;
        sum=sum+i;
    }

    cout << "i=" << i << endl;
    cout << " sum=" << sum << endl;

    return 0;
}
```

Microsoft Visual Studio 调试控制台

```
i=1000
sum=0
```


§. 基础知识题 – 循环结构



此页不要删除，也没有意义，仅仅为了分隔题目



§. 基础知识题 – 循环结构

3、观察程序运行结果

A. 观察下列程序的运行结果，回答问题并将程序的运行结果截图贴上(如果有错则贴错误信息截图)

```
#include <iostream>
#include <iomanip>    //格式输出
#include <cmath>      //fabs
#include <windows.h>  //取系统时间
using namespace std;
```

```
int main()
{
```

```
    int s=1;
    double n=1, t=1, pi=0;
```

```
    LARGE_INTEGER tick, begin, end;
    QueryPerformanceFrequency(&tick);    //取计数器频率
    QueryPerformanceCounter(&begin);      //取初始硬件定时器计数
```

```
    while(fabs(t)>1e-6) {
        pi=pi+t;
        n=n+2;
        s=-s;
        t=s/n;
    }
```

```
    QueryPerformanceCounter(&end); //获得终止硬件定时器计数
```

```
    pi=pi*4;
    cout << "n=" << setprecision(10) << n << endl;
    cout<<"pi="<<setiosflags(ios::fixed)<<setprecision(9)<<pi<< endl;
```

```
    cout << "计数器频率: " << tick.QuadPart << "Hz" << endl;
    cout << "时钟计数 : " << end.QuadPart - begin.QuadPart << endl;
    cout << setprecision(6) << (end.QuadPart - begin.QuadPart)/double(tick.QuadPart) << "秒" << endl;
```

```
    return 0;
}
```

用下面的迭代公式求Pi的值

$$1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \dots$$

(1) n, t, pi为double型

精度为1e-6: n=1000001	pi=3.141590654	时间=0.002179(秒)
1e-7: n=10000001	pi=3.141592454	时间=0.024990(秒)
1e-8: n=100000001	pi=3.141592634	时间=0.245191(秒)
1e-9: n=1000000001	pi=3.141592652	时间=2.127709(秒)

(因为机器配置不同, 时间值可能不同)

本页结果手填

(2) n, t, pi为float型

精度为1e-6: n=1000001	pi=3.141593933	时间=0.020859(秒)
1e-7: n=10000001	pi=3.141596556	时间=0.190094(秒)
1e-8: n= ?	pi= ?	时间= ?(秒)

问: 7项中哪个没结果? 为什么?

答: 精度为1e-8, n, t, pi为float型时没有结果。因为t很小时, 默认为以科学计数法形式存储, 又编译器默认为六位小数, 故加上小数点前的一位, 通过科学计数法可以表示7位数字, 但无法表示8位数字, fabs(t)>1e-8永真, 程序进入死循环, 故无结果。



§. 基础知识题 – 循环结构

3、观察程序运行结果

B. 观察下列程序的运行结果，回答问题并将程序的运行结果截图贴上(如果有错则贴错误信息截图)

#include <iostream> #include <iomanip>	打印100-200之间的素数	(1) 目前输出结果：一共21个，每10个一行
<pre>test1.cpp × test (全局范围) main() 1 #include <iostream> 2 #include <iomanip> 3 using namespace std; 4 int main() 5 { 6 int n = 0, i, m, k; 7 bool prime; 8 for (m = 103; m <= 200; m += 2) { //偶数没必要判断 9 prime = true; //对每个数，先认为是素数 10 k = int(sqrt(m)); 11 for (i = 2; i <= k; i++) 12 if (m % i == 0) { 13 prime = false; 14 break; 15 } 16 17 if (prime) { 18 cout << setw(5) << m; 19 n = n + 1; //计数器，只为了加输出换行 20 } 21 22 if (n % 10 == 0) { //每10个数输出一行 23 if (prime) //当m是素数时再换行 24 cout << endl; 25 } 26 } //end of for 27 28 return 0; 29 }</pre>		
		(2) 将m的初值从101改为103，应该是20个，共2行 实际呢？为什么？
		答：实际是20个，共4行。 1、因为151、157之间隔了153、155两个数，而153、155不是素数，根据代码需break出内层的那个for循环。 2、此时prime=false，不执行下一个if判断语句，导致计数器没有加1，仍为10。 3、故继续执行再下一个if判断语句，输出了换行符，且因为有两个非素数，上述过程进行了两次。
		(3) 将左侧程序改正确 (正确程序贴图在左侧，覆盖现有内容即可)

§. 基础知识题 – 循环结构



此页不要删除，也没有意义，仅仅为了分隔题目