

# §. 基础知识题 – 循环结构



要求:

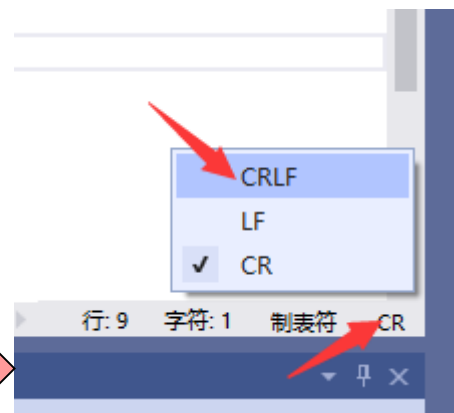
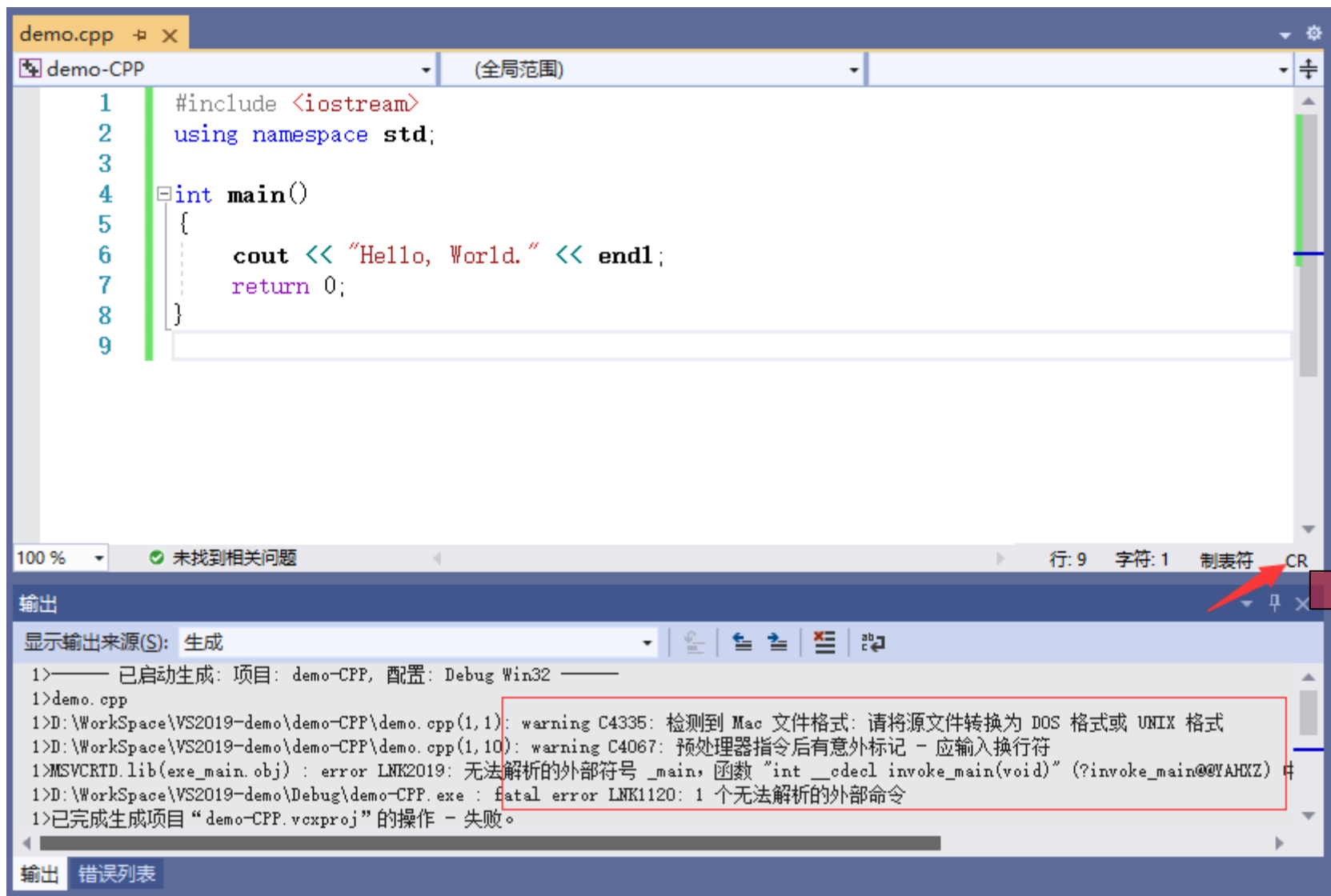
- 1、完成本文档中所有的题目并按要求写出分析、运行结果（包括截图）
- 2、无特殊说明，均使用VS2019编译即可
- 3、直接在本文件上作答，**写出答案/截图（不允许手写、手写拍照截图）**即可；填写答案时，为适应所填内容或贴图，**允许调整**页面的字体大小、颜色、文本框的位置等(在保证一页一题的前提下，具体页面布局可以自行发挥，简单易读即可)
- 4、转换为pdf后提交
- 5、**10月25日前**网上提交本次作业（在“实验报告”中提交）
- 6、作业部分细节内容可能会有调整，随时注意Update!!!



## §. 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



特别提醒:

- 1、前几次的文档里作业中，发现部分同学贴图是直接贴全屏，  
导致有效内容根本无法阅读
- 2、如果该部分内容无法阅读，该题是不能得分的!!!
- 3、一个不用强调的规则是：只贴有效部分，并且内容须清晰可辨



## §. 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)

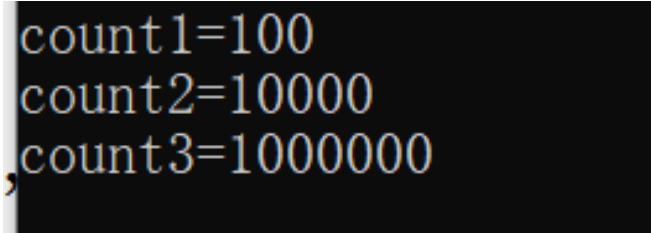
```
#include <iostream>
using namespace std;

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



```
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？

当外层循环执行一次时，内层循环执行到结束为止，之后外层循环才能执行第二次。而这就造成了题中，外层执行1次，内层执行100次，外层执行100次，内层执行10000次。



## §. 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

```
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？

当外层循环执行一次时，内层循环执行到结束为止，之后外层循环才能执行第二次。而此题中外层循环开始时变量有联系，所以内层循环开始的时候不是从1开始，而是从外层循环进行到第几次的个数执行。

## §. 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()
{
```

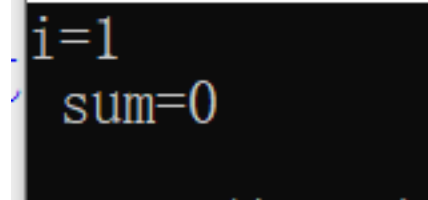
```
    int i=0, sum=0;
```

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

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

```
    return 0;
```

```
}
```



i=1  
sum=0

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

```
int main()
{
```

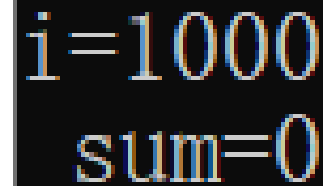
```
    int i=0, sum=0;
```

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

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

```
    return 0;
```

```
}
```



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=\_0.021806\_(秒)  
1e-7: n=\_\_\_10000001\_\_\_ pi=\_3.141592454\_ 时间=\_0.131354\_(秒)  
1e-8: n=\_\_\_100000001\_\_\_ pi=\_3.141592634\_ 时间=\_1.133172\_(秒)  
1e-9: n=\_\_\_1000000001\_\_\_ pi=\_3.141592652\_ 时间=\_9.016091\_(秒)

(因为机器配置不同，时间值可能不同)

(2) n, t, pi为float型

精度为1e-6: n=\_\_\_1000001\_\_\_ pi=\_3.141593933\_ 时间=\_0.046685\_(秒)  
1e-7: n=\_\_\_10000001\_\_\_ pi=\_3.141596556\_ 时间=\_0.455519\_(秒)  
1e-8: n=\_\_\_\_\_ pi=\_\_\_\_\_ 时间=\_\_\_\_\_ (秒)

问：7项中哪个没结果？为什么？

最后一个无结果，float的精度仅限于6位有效数字数字在达到小数点6位以后数字就不可信了，在精度达到10的-8次幂时数字就不那么发生变化了，导致，最后绝对值一直在循环合理范围内，不出循环，运行很长时间也不出循环，所以会导致一直显示不出来结果。



## §. 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>
using namespace std;
int main()
{
    int n = 0, i, m, k;
    bool prime;
    for (m = 101; m <= 200; m += 2) { //偶数没必要判断
        prime = true;                //对每个数，先认为是素数
        k = int(sqrt(m));
        for (i = 2; i <= k; i++)
            if (m % i == 0) {
                prime = false;
                break;
            }
        if (prime) {
            cout << setw(5) << m;
            n = n + 1;                //计数器，只为了加输出换行
            if (n % 10 == 0)         //每10个数输出一行
                cout << endl;
        }
    } //end of for
    return 0;
}
```

(1) 目前输出结果：一共21个，每10个一行

Microsoft Visual Studio 调试控制台

```
101  103  107  109  113  127  131  137  139  149
151  157  163  167  173  179  181  191  193  197
199
```

(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？

```
103  107  109  113  127  131  137  139  149  151
157  163  167  173  179  181  191  193  197  199
```

之所以第一次101表面看起来没有问题，第二次103看起来就有问题，是因为149下一个素数是紧挨着的151，而151下一个紧挨着的153不是素数，因为149下一个151是素数，则prime是真，就会执行对应的if(prime)，n就是11就不会输入换行，而151下面的153的时候prime为假，直接又执行一次换行。

(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