

# § . 基础知识题 - 关系运算、逻辑运算与选择结构



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VS2022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、**3月28日前**网上提交本次作业（在“文档作业”中提交）



## §. 基础知识题 - 关系运算、逻辑运算与选择结构

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

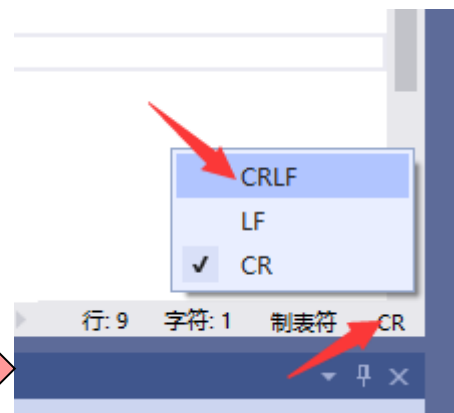
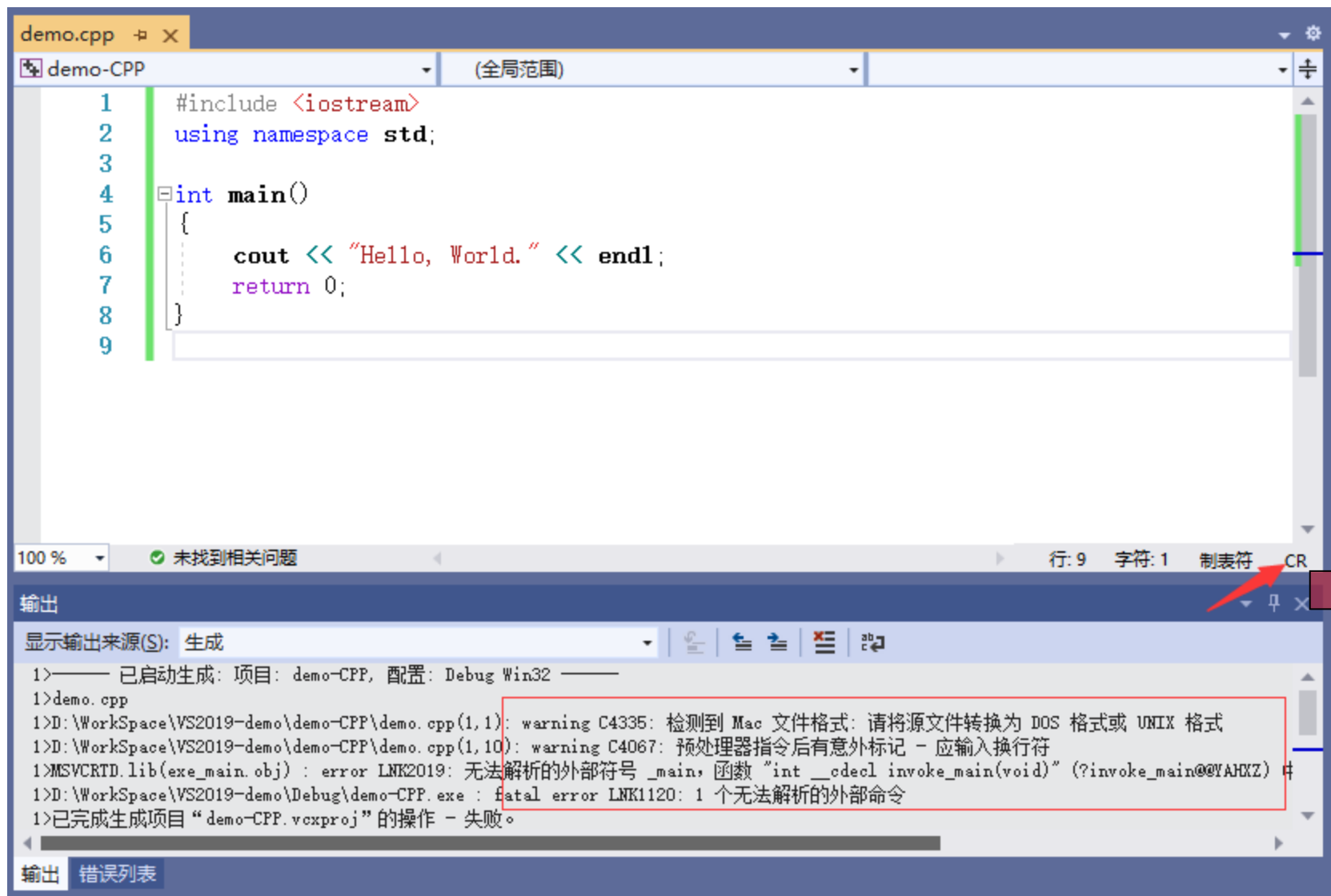
```
Microsoft Visual Studio 调试控制台  
Hello, world!
```



## §. 基础知识题 - 关系运算、逻辑运算与选择结构

附：用WPS等其他第三方软件打开PPT，将代码复制到VS2022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 - 关系运算、逻辑运算与选择结构



特别提示:

- ★ 本次作业的答案，除特别提示外，上课全讲过，课件上都有!!!
- ★ 作业本质就是对上课内容及课件的review(因为读懂程序的逻辑很重要)
- ★ 对上课接受程度较好的同学，可能有点重复/多余，但还得做



## §. 基础知识题 - 关系运算、逻辑运算与选择结构

### 1、关系运算符的求值顺序

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 a=1, b=2, c=3, d;

    d = a > b > c;
    cout << d << endl;

    d = a < b < c;
    cout << d << endl;

    d = b > a < c;
    cout << d << endl;

    return 0;
}
```

#### 1、贴运行结果

Microsoft Visual Studio 调试控制台

```
0
1
1
```

```
warning C4804: ">" : 在操作中使用类型 "bool" 不安全
warning C4804: "<" : 在操作中使用类型 "bool" 不安全
warning C4804: "<" : 在操作中使用类型 "bool" 不安全
```

#### 2、VS下为什么会有三个warning? 说说你的理解

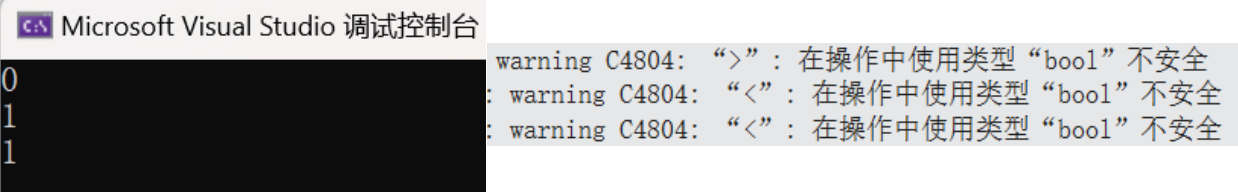
在第一次比较运算中，返回值为bool类型的0和1，那么第二次运算就是0或1与第三个数比较运算而非第二个数与第三个进行比较运算，可能会产生错误结果，所以计算机会对第二次比较运算报warning。



# §. 基础知识题 - 关系运算、逻辑运算与选择结构

## 1、关系运算符的求值顺序

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)

<pre>#include &lt;iostream&gt; using namespace std;  int main() {     int a=3, b=2, c=1, d;      d = a &gt; b &gt; c;     cout &lt;&lt; d &lt;&lt; endl;      d = a &lt; b &lt; c;     cout &lt;&lt; d &lt;&lt; endl;      d = b &gt; a &lt; c;     cout &lt;&lt; d &lt;&lt; endl;      return 0; }</pre>	<div>1、贴运行结果</div> <div></div> <div>2、<math>a &gt; b &gt; c</math>这个式子，按人的常规理解(<math>3 &gt; 2</math>且<math>2 &gt; 1</math>)是正确的，为什么结果是0 ? <math>a &lt; b &lt; c</math>这个式子，按人的常规理解(<math>3 &lt; 2</math>且<math>2 &lt; 1</math>)是错误的，为什么结果是1 ? <math>b &gt; a &lt; c</math>这个式子，按人的常规理解(<math>2 &gt; 3</math>且<math>3 &lt; 1</math>)是错误的，为什么结果是1 ? (文字简单说明即可)</div> <div>1) 首先进行<math>a &gt; b</math>计算，正确，返回值为1，接着进行返回值<math>1 &gt; c</math>计算，错误，则输出为0; 2) 首先进行<math>a &lt; b</math>计算，错误，返回值为0，接着进行返回值<math>0 &lt; c</math>计算，正确，则输出为1; 3) 首先进行<math>b &gt; a</math>计算，错误，返回值为0，接着进行返回值<math>0 &lt; c</math>计算，正确，则输出为1;</div>
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------



## §. 基础知识题 – 关系运算、逻辑运算与选择结构

此页不要删除，也没有意义，仅仅为了分隔题目



# §. 基础知识题 - 关系运算、逻辑运算与选择结构

## 2、关系运算符与实数

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)

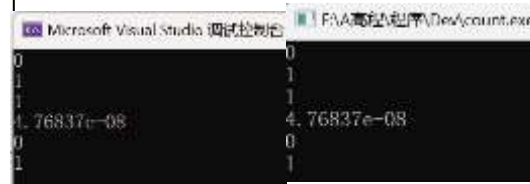
```
#include <iostream>
#include <cmath>
using namespace std;

int main()
{
    float f1 = 100.25;
    cout << (f1 - 100.25) << endl;
    cout << (f1 == 100.25) << endl;
    cout << (fabs(f1-100.25) < 1e-6) << endl;

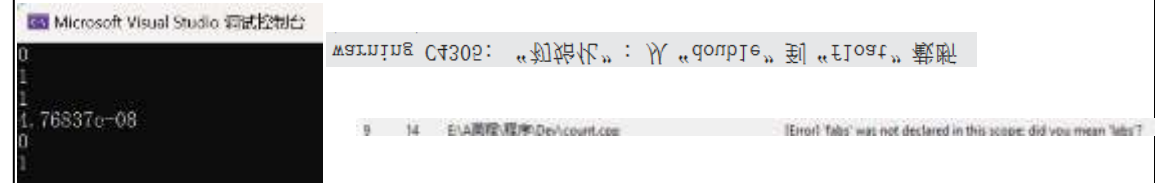
    float f2 = 1.2;
    cout << (f2 - 1.2) << endl;
    cout << (f2 == 1.2) << endl;
    cout << (fabs(f2-1.2) < 1e-6) << endl;

    return 0;
}
```

### 1、贴VS+Dev下的运行结果



### 2、删除第2行的#include<cmath>后，再次贴VS+Dev的运行结果



### 3、由本例得出的结论，实数进行相等比较时的通用方法是 \_判断二者的差的绝对值是否小于特定数量级\_\_\_\_\_





## §. 基础知识题 - 关系运算、逻辑运算与选择结构

### 2、关系运算符与实数

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 <cmath> //VS可不加
using namespace std;

int main()
{
    double d1=123.456789012345678;
    double d2=123.456789123456789;
    cout << (d1==d2) << endl;
    cout << (fabs(d1-d2)<1e-6) << endl;
    cout << (fabs(d1-d2)<1e-7) << endl;

    float f1=123.456789012345678;
    float f2=123.456789123456789;
    cout << (f1==f2) << endl;
    cout << (fabs(f1-f2)<1e-6) << endl;
    cout << (fabs(f1-f2)<1e-7) << endl;

    return 0;
} //VS有两个warning
```

#### 1、贴运行结果

Microsoft Visual Studio 调试控制台

warning C4305: “初始化”: 从“double”到“float”截断  
warning C4305: “初始化”: 从“double”到“float”截断

#### 2、观察fabs(\*\*)<1e-6 和 fabs(\*\*)<1e-7在float和double下的表现，哪个相同？哪个不同？为什么？

fabs(\*\*)<1e-6相同， fabs(\*\*)<1e-7不相同

因为double精度更高，可以得到更精准的结果；而float有效位数较少，对于有效位数较多的数存在储存不准确的情况，储存时，f1和f2可能储存为同一个二进制数。



## §. 基础知识题 - 关系运算、逻辑运算与选择结构

此页不要删除，也没有意义，仅仅为了分隔题目



## §. 基础知识题 - 关系运算、逻辑运算与选择结构

### 3、逻辑常量与逻辑变量

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()
{
    cout << true    << endl;
    cout << "true"  << endl;

    cout << endl;

    cout << false   << endl;
    cout << "false" << endl;

    return 0;
}
```

#### 1、贴运行结果



#### 2、解释 true 和 "true" 的区别 (false和"false")

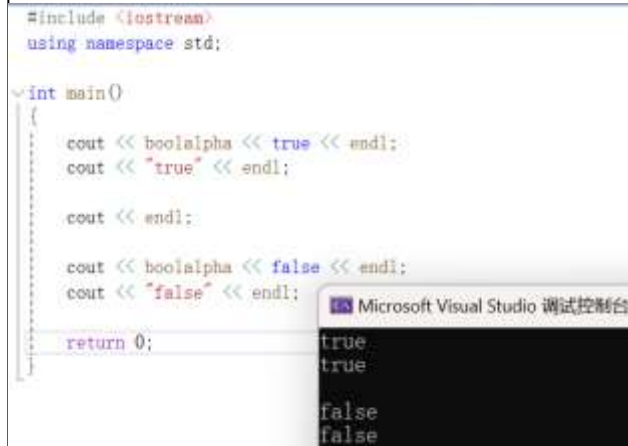
True/false为bool类型常量，而 "true" / "false" 为字符串

3、进阶思考：目前直接输出逻辑常量true和false，在屏幕上输出的输出是1/0，如果想输出为true/false，应该怎么做？

**注意：**1、不允许用分支语句/条件运算符

2、提示：去网上查一个前导格式控制符(课件无)

可以在输出bool前加boolalpha。





## §. 基础知识题 - 关系运算、逻辑运算与选择结构

### 3、逻辑常量与逻辑变量

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()
{
    bool k1 = true;

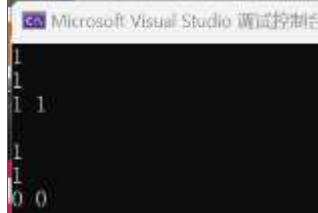
    cout << sizeof(true) << endl;
    cout << sizeof(k1) << endl;
    cout << k1 << ' ' << int(k1) << endl;

    cout << endl;

    bool k2 = false;
    cout << sizeof(false) << endl;
    cout << sizeof(k2) << endl;
    cout << k2 << ' ' << int(k2) << endl;

    return 0;
}
```

#### 1、贴运行结果



2、bool型常量/变量在内存中占用\_\_1\_\_字节，值是\_0或1\_\_

总结bool型常量/变量在输出时的规则

(限制：在无3.A的前导格式控制符的前提下)

bool型常量/变量在输出时只会输出0或1（没有控制符）；若涉及类型转化，按整形常量处理。



## §. 基础知识题 - 关系运算、逻辑运算与选择结构

### 3、逻辑常量与逻辑变量

C. 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)panduan

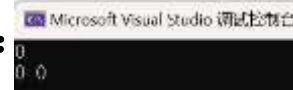
```
#include <iostream>
using namespace std;

int main()
{
    bool k;

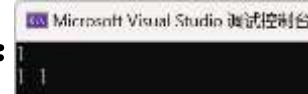
    cin >> k;
    cout << k << ' ' << int(k) << endl;

    return 0;
}
```

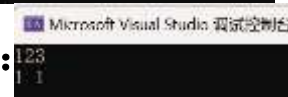
1、输入0，输出是：



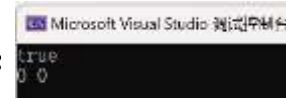
2、输入1，输出是：



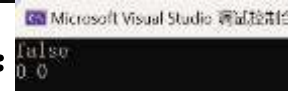
3、输入123，输出是：



4、输入true，输出是：



5、输入false，输出是：



总结bool型变量在输入时的规则：

bool型变量输入时按“非0为真0为假”的原则，输入非零数则判断为真，输入1，输入0则判断为假，输入0；当数字前有非数字字符时，则输入0.



## §. 基础知识题 - 关系运算、逻辑运算与选择结构

### 3、逻辑常量与逻辑变量

D. 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()
{
    bool k;

    k='A';
    cout << k << ' ' << (int)k << endl;

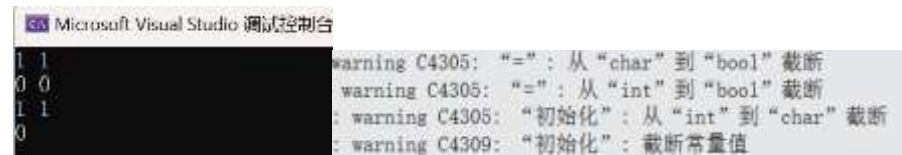
    k=0;
    cout << k << ' ' << (int)k << endl;

    k=256;
    cout << k << ' ' << (int)k << endl;

    char c = 256;
    cout << (int)c << endl;

    return 0;
}
```

1、贴运行结果



2、解释VS下warning的意思

由于bool只能储存0和1，当其他类型的非0或1数赋值给bool类型时，就会产生截断；char类型上限为255. 将256赋给c时，会产生截断

3、k='A' 是1字节赋值给1字节，为什么还有warning?

因为bool类型数只有0和1，赋值时会有截断，产生warning

4、k=256如果按整型的4字节赋给1字节，k应该是多少？

现在实际是多少？为什么？(和c对比)

应该为0实际为1；因为bool赋值时，按非0为真0为假的原则，不是多字节赋少字节的原则

5、为什么不 cout << c，而是 (int)c ？

ASCII为0的字符输出结果为空字符，但输出结果为空字符的字符对应ASCII码不一定为0，转为int型后，更容易判断256赋给char型之后的结果

6、“非0为真0为假”这句话如何解释？

对于bool型，若输入一个非0的结果，则判断为真，输出为1；若输入一个0，则判断为假，输出为0.



## §. 基础知识题 - 关系运算、逻辑运算与选择结构

### 3、逻辑常量与逻辑变量

E. 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()
{
    bool f=true;
    int a=10;

    a=a+f;
    cout << a << endl;

    return 0;
}
```

1、贴运行结果

Microsoft Visual Studio 调试控制台

11

2、当bool参与表达式计算时，当做\_\_一个整型常量1或0\_\_



## §. 基础知识题 - 关系运算、逻辑运算与选择结构

此页不要删除，也没有意义，仅仅为了分隔题目





## §. 基础知识题 – 关系运算、逻辑运算与选择结构

### 4、逻辑运算符与逻辑运算

A. 完成下列两个表格的填写（a/b是两个逻辑值，**填写的内容不要用黑色**）

a	b	!a	!b	a&&b	a  b
1	1	0	0	1	1
1	0	0	1	0	1
0	1	1	0	0	1
0	0	1	1	0	0

a	b	!a	!b	a&&b	a  b
非0	非0	0	0	1	1
非0	0	0	1	0	1
0	非0	1	0	0	1
0	0	1	1	0	0



## §. 基础知识题 - 关系运算、逻辑运算与选择结构

### 4、逻辑运算符与逻辑运算

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 a=1, b=2, c=3, d=4, m=1, n=1;

    cout << "m=" << m << " n=" << n << endl;
    (m=a>b)&&(n=c>d);
    cout << "m=" << m << " n=" << n << endl;

    return 0;
}
```

1、贴运行结果

```
Microsoft Visual Studio 调试控制台
m=1 n=1
m=0 n=1
```

2、解释  $(m=a>b)\&\&(n=c>d)$  的求值过程 (标出步骤顺序)

步骤1:  $a>b$  结果: 0

步骤2:  $m=0$  结果:  $m(0)$

步骤3:  $m\&\&(n=c>d)$  结果: 0

3、短路运算的意思是: \_仅当必须执行下一逻辑运算符才能求出解时, 才执行该运算符, 否则不执行\_



## §. 基础知识题 – 关系运算、逻辑运算与选择结构

### 4、逻辑运算符与逻辑运算

C. 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 a = 1, b=2;
    cout << "a=" << a << " b=" << b
    << endl;
    5>3 && 2 || (a=8<4 - !0), b= 8<4;
    cout << "a=" << a << " b=" << b
    << endl;
    return 0;
}
```

C:\> Microsoft Visual Studio 调试控制台

```
a=1 b=2
a=1 b=0
```

有以下逗号表达式，其表达式1是逻辑表达式，表达式2按需构造  
5>3 && 2 || 8<4 - !0, \*\*\*

1、构造一个测试程序，在不改变该表达式目前求值顺序的情况下（允许插入新的运算，但目前这几个运算符的顺序不要变），证明两点：

- 1、8<4 - !0 存在短路运算
- 2、\*\*\* 不存在短路运算

2、用栈方式画包含短路运算的表达式，则从分析到短路运算符进栈开始（本例中为||），忽略\_\_比||优先级高的\_\_运算符。  
（所有 / 比||优先级高的）



## §. 基础知识题 – 关系运算、逻辑运算与选择结构

此页不要删除，也没有意义，仅仅为了分隔题目



## §. 基础知识题 - 关系运算、逻辑运算与选择结构

### 5、if语句 - 基本使用

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;

    cout<<"请输入成绩[0-100]"<<endl;
    cin >> i;

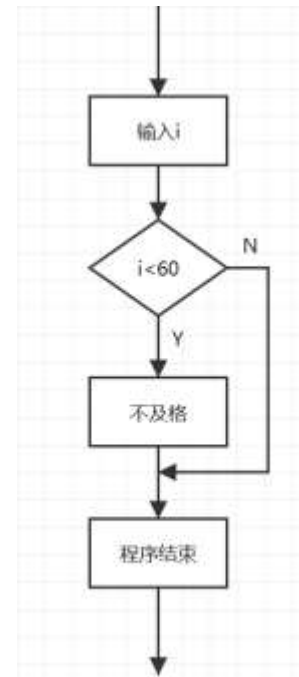
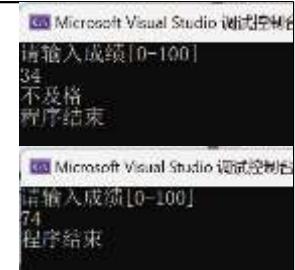
    if (i<60) {
        cout << "不及格" << endl;
    }
    cout << "程序结束" << endl;

    return 0;
}
```

1、输入34，贴运行结果

2、输入74，贴运行结果

3、画出程序对应的流程框图





## §. 基础知识题 - 关系运算、逻辑运算与选择结构

### 5、if语句 - 基本使用

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;

    cout<<"请输入成绩[0-100]"<<endl;
    cin >> i;

    if (i<60) {
        cout << "不及格" << endl;
        cout << "程序结束" << endl; //未缩进
    }

    return 0;
}
```

1、输入34，贴运行结果

2、输入74，贴运行结果

3、画出程序对应的流程框图

4、程序标注“未缩进”的行，\_应该\_\_\_\_\_（应该/不应该）缩进





## §. 基础知识题 - 关系运算、逻辑运算与选择结构

### 5、if语句 - 基本使用

C. 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;

    cout<<"请输入成绩[0-100]"<<endl;
    cin >> i;

    if (i<60;) {
        cout << "不及格" << endl;
        cout << "程序结束" << endl; //未缩进
    }

    return 0;
}
```

贴编译错误并给出解释

在if后表达式后括号内表达式后添加了分号，使表达式变成了语句。

```
warning C4552: " < ": 未使用表达式结果
error C2429: 语言功能 "if/switch 中的 init-statement" 需要编译器标志 "/std:c++17"
error C2059: 语法错误: ")"
error C2143: 语法错误: 缺少 ";" (在 "{" 的前面)
```



## §. 基础知识题 - 关系运算、逻辑运算与选择结构

### 5、if语句 - 基本使用

#### D. 观察下列程序的运行结果，回答问题

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

int main()
{
    int i;
    cout << "请输入成绩[0-100]" << endl;
    cin >> i;
    if (i>=90 && i<=100)
        cout << "优" << endl;
    else if (i>=80 && i<90)
        cout << "良" << endl;
    else if (i>=70 && i<80)
        cout << "中" << endl;
    else if (i>=60 && i<70)
        cout << "及格" << endl;
    else if (i>=0 && i<60)
        cout << "不及格" << endl;
    else
        cout << "输入错误" << endl;
    cout << "程序结束" << endl;
    return 0;
}
```

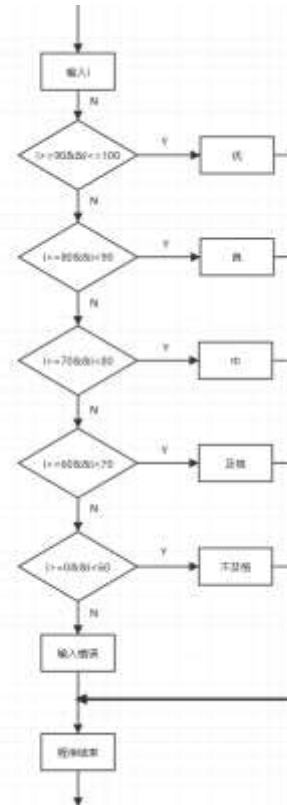
1、给出程序的流程框图(注意字体的清晰可辨)

2、 $i < 90$ 能否改为 $i \leq 89$ ? 哪个更好?

可以。 $i < 90$ 更好, 当成绩可以有小数时, 修改更方便。

3、 $i < 90$ 能否改为 $i \leq 90$ ? 运行是否正确?

可以, 运行正确。







## §. 基础知识题 - 关系运算、逻辑运算与选择结构

此页不要删除，也没有意义，仅仅为了分隔题目



# §. 基础知识题 - 关系运算、逻辑运算与选择结构

## 6、if语句 - 多重嵌套

A. 一个有10行代码的if语句嵌套，回答问题

0: if (表达式) { 1: if (表达式) { 2: } 3: else { 4: } 5: } 6: else { 7: if (表达式) { 8: } 9: }	第0行的"{" 和 第_5__行的"}"配对  第1行的"{" 和 第_2__行的"}"配对  第3行的"{" 和 第_4__行的"}"配对  第6行的"{" 和 第_9__行的"}"配对  第7行的"{" 和 第_8__行的"}"配对   总结：给出大括号配对的基本准则 自上而下，忽略{, 以}为准向上匹配未配对的{
-------------------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------



## §. 基础知识题 - 关系运算、逻辑运算与选择结构

### 6、if语句 - 多重嵌套

B. 一个if语句嵌套如下，回答问题

```
if (表达式1) {  
    if (表达式2) {  
        A;  
    }  
    B;  
}
```

1、当表达式1\_\_真\_\_(真/假/任意)，表达式2\_\_真\_\_(真/假/任意)时，

执行语句A

2、当表达式1\_\_真\_\_(真/假/任意)，表达式2\_\_任意\_\_(真/假/任意)时，

执行语句B



## §. 基础知识题 - 关系运算、逻辑运算与选择结构

### 6、if语句 - 多重嵌套

C. 一个if语句嵌套如下，回答问题

```
if (表达式1) {  
    if (表达式2) {  
        A;  
    }  
    else {  
        B;  
    }  
    C;  
}  
else {  
    if (表达式3) {  
        D;  
    }  
    E;  
}
```

1、当表达式1\_\_真\_\_(真/假/任意)，表达式2\_\_真\_\_(真/假/任意)时，执行语句A

2、当表达式1\_\_真\_\_(真/假/任意)，表达式2\_\_假\_\_(真/假/任意)时，执行语句B

3、当表达式1\_\_真\_\_(真/假/任意)，表达式2\_\_任意\_\_(真/假/任意)时，执行语句C

4、当表达式1\_\_假\_\_(真/假/任意)，表达式3\_\_真\_\_(真/假/任意)时，  
执行语句D

5、当表达式1\_\_假\_\_(真/假/任意)，表达式3\_\_任意\_\_(真/假/任意)时，  
执行语句E



## §. 基础知识题 - 关系运算、逻辑运算与选择结构

### 6、if语句 - 多重嵌套

D. 一个if语句嵌套如下，回答问题

```
if (表达式1) {  
    if (表达式2) {  
        A;  
    }  
    else {  
        B;  
    }  
    C;  
}  
→ F;  
else {  
    if (表达式3) {  
        D;  
    }  
    E;  
}
```

在6.C的基础上，在箭头位置插入语句F

1、请构造一个符合此要求的测试程序，并给出该程序的程序及编译错误截图

8 cin >> a;  
9  
10 if (a>0) {  
11 if (int(a)==a) {  
12 cout << "这是一个正的整数" << endl;  
13 }  
14 else {  
15 cout << "这是一个正的小数" << endl;  
16 }  
17 cout<<"正数"<<endl;  
18 }  
19 cout<<"输出结束"<<endl;  
20 else {  
21 if (a==0) {  
22 cout << "这是零" << endl;  
23 }  
24 cout << "非正数" << endl;  
25 }  
26  
27 return 0;  
28 }

error C2181: 没有匹配 if 的非法 else

2、请说明错误原因

If/else整个双支语句可以看成一个语句，中间不允许插入任何其他语句



§ . 基础知识题 – 关系运算、逻辑运算与选择结构

6、if语句 – 多重嵌套

E. 一个if语句嵌套如下，回答问题

<pre>if (表达式1) {   if (表达式2) {     A;   }   B; } else {   C; }</pre>	<p>左侧代码按缩进格式排版</p> <pre>if (表达式1) {     if (表达式2) {         A;     }     B; } else {     C; }</pre>	<p>1、当表达式1__真__(真/假/任意)，表达式2__真__(真/假/任意)时，执行语句A</p> <p>2、当表达式1_真____(真/假/任意)，表达式2__任意__(真/假/任意)时，执行语句B</p> <p>3、当表达式1__假__(真/假/任意)，表达式2__任意__(真/假/任意)时，     执行语句C</p>
<pre>if (表达式1) {   if (表达式2) {     A;   }   else {     B;   }   C; }</pre>	<p>左侧代码按缩进格式排版</p> <pre>if (表达式1) {     if (表达式2) {         A;     }     else {         B;     }     C; }</pre>	<p>1、当表达式1__真__(真/假/任意)，表达式2__真__(真/假/任意)时，执行语句A</p> <p>2、当表达式1__真__(真/假/任意)，表达式2__假__(真/假/任意)时，执行语句B</p> <p>3、当表达式1__真__(真/假/任意)，表达式2__任意__(真/假/任意)时，执行语句C</p>



## §. 基础知识题 - 关系运算、逻辑运算与选择结构

此页不要删除，也没有意义，仅仅为了分隔题目



## §. 基础知识题 - 关系运算、逻辑运算与选择结构

### 7、条件运算符与条件表达式

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 a,b;

    cin >> a >> b;

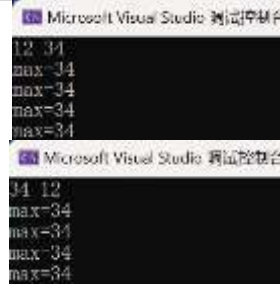
    if (a>b)
        cout << "max=" << a << endl;
    else
        cout << "max=" << b << endl;

    a > b ? cout << "max=" << a << endl : cout << "max=" << b << endl; //1

    cout << "max=" << (a>b?a:b) << endl; //2

    printf("max=%d", a>b?a:b); //3

    return 0;
}
```



1、输入12 34，给出运行截图

2、输入34 12，给出运行截图

3、//1 //2 //3这三种条件运算符的使用，按你的喜欢程度排序为\_ //2 //3 //1\_\_





# §. 基础知识题 - 关系运算、逻辑运算与选择结构

## 7、条件运算符与条件表达式

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)

<pre>#include &lt;iostream&gt; using namespace std;  error C2678: 二进制 “?” : 没有找到接受 “std::basic_ostream&lt;char, std::char_traits&lt;char&gt;&gt;” 类型的左操作数的运算符(或没有可接受的转换)  int main() {     int a=1, b=2;      a==1 ? "Hello" : 123; //编译报错      a&gt;b ? cout &lt;&lt; a : printf("%d", b); //编译报错      a==1 ? 'A' : 123; //编译正确      return 0; }</pre>	<p>1、给出编译报错的截图</p> <p>error C2446: “:” : 没有从 “int” 到 “const char [6]” 的转换</p> <p>2、条件表达式使用的三句中，前两句报错，最后一句正确，总结下条件表达式使用时的限制规则(提示：注意表达式2和表达式3的类型)</p> <p>条件运算的表达式2和表达式3不能为字符串，也不能为返回值为非整数的表达式，只能为字符常量，数字，返回值为整数的表达式；表达式1、2、3雷星可以不同，但表达式2、3类型必须相容。</p>
---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------



## §. 基础知识题 - 关系运算、逻辑运算与选择结构

此页不要删除，也没有意义，仅仅为了分隔题目



# §. 基础知识题 - 关系运算、逻辑运算与选择结构

## 8、switch-case语句

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 score;
    cout<<"请输入成绩[0-100]"<<endl;
    cin >> score;
    switch(score/10) {
        case 10:
        case 9:
            cout<<"优"<<endl;
            break;
        case 8:
            cout<<"良"<<endl;
            break;
        case 7:
            cout<<"中"<<endl;
            break;
        case 6:
            cout<<"及格"<<endl;
            break;
        case 5:
        case 4:
        case 3:
        case 2:
        case 1:
        case 0:
            cout<<"不及格"<<endl;
            break;
        default:
            cout<<"输入错误"<<endl;
            break;
    }

    return 0;
}
```

程序的期望，是当输入的score在[0..100]时，分段输出“优/良/中/及格/不及格”，否则输出“输入错误”

1、程序不完全正确，找出不符合期望的两个数据区间并给出运行截图  
(不需要改对)

不符合期望的两个数据区间为 (100, 110) , (-10, 0)

请输入成绩[0-100] 101 优	请输入成绩[0-100] 102 优	请输入成绩[0-100] 103 优	请输入成绩[0-100] 104 优	请输入成绩[0-100] 105 优	请输入成绩[0-100] 106 优	请输入成绩[0-100] 107 优
请输入成绩[0-100] 108 优	请输入成绩[0-100] 109 优	请输入成绩[0-100] -1 不及格	请输入成绩[0-100] -2 不及格	请输入成绩[0-100] -3 不及格	请输入成绩[0-100] -4 不及格	请输入成绩[0-100] -5 不及格
请输入成绩[0-100] -6 不及格	请输入成绩[0-100] 7 不及格	请输入成绩[0-100] 8 不及格	请输入成绩[0-100] -9 不及格			



# §. 基础知识题 - 关系运算、逻辑运算与选择结构

## 8、switch-case语句

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()
{
    const int k=5;
    int score;
    cout<<"请输入成绩[0-100]"<<endl;
    cin >> score;
    switch(score/10) {
        case 10:
        case 9:
            cout<<"优"<<endl;
            break;
        case 6:
            cout<<"及格"<<endl;
            break;
        default:
            cout<<"输入错误"<<endl;
            break;
        case k+2:
            cout<<"中"<<endl;
            break;
        case 8:
            cout<<"良"<<endl;
            break;
        case 5:
        case 4:
        case 3:
        case 2:
        case 1:
        case 0:
            cout<<"不及格"<<endl;
            break;
    }

    return 0;
}
```

在8. A的基础上  
1、将6、8、default的位置进行了交换  
2、将7写为常变量+常量形式

验证此程序与8. A的功能是否完全一致  
(即：8. A中正确的, 此程序中同样正确；8. A错误的，此程序中同样错误)



结论：8. A和8. B\_完全一致\_\_\_\_\_ (完全一致/不完全一致)  
如果不完全一致，给出表现不一致的测试数据的运行截图



## §. 基础知识题 - 关系运算、逻辑运算与选择结构

### 8、switch-case语句

C. 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 k=5;
    int score;
    cout<<"请输入成绩[0-100]"<<endl;
    cin >> score;
    switch(score/10) {
        case 10:
        case 9:
            cout<<"优"<<endl;
            break;
        case 6:
            cout<<"及格"<<endl;
            break;
        default:
            cout<<"输入错误"<<endl;
            break;
        case k+2:
            cout<<"中"<<endl;
            break;
        case 8:
            cout<<"良"<<endl;
            break;
        case 5:
        case 4:
        case 3:
        case 2:
        case 1:
        case 0:
            cout<<"不及格"<<endl;
            break;
    }

    return 0;
}
```

在8.B的基础上，将k从const int改为int

1、给出编译错误的截图

error C2131: 表达式的计算结果不是常数  
error C2051: case 表达式不是常量

2、解释错误原因

Switch case中的case只能跟整型常量或常量表达式，不能跟变量。



## §. 基础知识题 - 关系运算、逻辑运算与选择结构

### 8、switch-case语句

D. 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 score;
    cout<<"请输入成绩[0-100]"<<endl;
    cin >> score;
    switch(score/10) {
        case 10:
        case 9:
            cout<<"优"<<endl;
            break;
        case 8:
            cout<<"良"<<endl;
            break;
        case 7:
            cout<<"中"<<endl;
            break;
        case 6:
        case 4+2:
            cout<<"及格"<<endl;
            break;
        case 5:
        case 4:
        case 3:
        case 2:
        case 1:
        case 0:
            cout<<"不及格"<<endl;
            break;
        default:
            cout<<"输入错误"<<endl;
            break;
    }

    return 0;
}
```

在8. A的基础上，多了一个case 4+2

1、给出编译错误的截图

error C2196: case 值“6”已使用

2、解释错误原因

任何两个case标签的值不能相同



# §. 基础知识题 - 关系运算、逻辑运算与选择结构

## 8、switch-case语句

E. 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)

<pre>#include &lt;iostream&gt; using namespace std; int main() {     float score;     cout&lt;&lt;"请输入成绩[0-100]"&lt;&lt;endl;     cin &gt;&gt; score;     switch(score/10) {         case 10:         case 9:             cout&lt;&lt;"优"&lt;&lt;endl;             break;         case 8:             cout&lt;&lt;"良"&lt;&lt;endl;             break;         case 7:             cout&lt;&lt;"中"&lt;&lt;endl;             break;         case 6:             cout&lt;&lt;"及格"&lt;&lt;endl;             break;         case 5:         case 4:         case 3:         case 2:         case 1:         case 0:             cout&lt;&lt;"不及格"&lt;&lt;endl;             break;         default:             cout&lt;&lt;"输入错误"&lt;&lt;endl;             break;     }      return 0; }</pre>	<p>在8.A的基础上，将score从int改为float</p> <p>1、给出编译错误的截图 error C2450: 类型为“float”的 switch 表达式无效</p> <p>2、解释错误原因 Switch语句只能用int、char、short、long、bool等整型，float、double等浮点型不能用于switch语句</p>
-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------





# §. 基础知识题 - 关系运算、逻辑运算与选择结构

## 8、switch-case语句

F. 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 score;
    cout<<"请输入成绩[0-100]"<<endl;
    cin >> score;
    switch(score/10) {
        case 10:
        case 9:
            cout<<"优"<<endl;
            break;
        case 8:
            cout<<"良"<<endl;

        case 7:
            cout<<"中"<<endl;
            break;
        case 6:
            cout<<"及格"<<endl;
            break;
        case 5:
        case 4:
        case 3:
        case 2:
        case 1:
        case 0:
            cout<<"不及格"<<endl;
            break;
        default:
            cout<<"输入错误"<<endl;
            break;
    }

    return 0;
}
```

在8.A的基础上，删除case 8后面的break

1、给出与8.A运行结果不一致的测试数据即截图

80、81、82、83、84、85、86、87、88、89

请输入成绩[0-100]	请输入成绩[0-100]	请输入成绩[0-100]	请输入成绩[0-100]	请输入成绩[0-100]	请输入成绩[0-100]	请输入成绩[0-100]
80	81	82	83	84	85	86
良	良	良	良	良	良	良
中	中	中	中	中	中	中

请输入成绩[0-100] 请输入成绩[0-100]

请输入成绩[0-100]	请输入成绩[0-100]
88	89
良	良
中	中

2、解释break的作用

结束该分支的执行，跳转至switch语句末尾。





# §. 基础知识题 - 关系运算、逻辑运算与选择结构

## 8、switch-case语句

G. 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 score;
    cout<<"请输入成绩[0-100]"<<endl;
    cin >> score;
    switch(score/10) {
        case 10:
        case 9:
            cout<<"优"<<endl;
            break;
        case 8:
            cout<<"良"<<endl;
            break;
        case 7:
            cout<<"中"<<endl;
            break;
        case 6:
            cout<<"及格"<<endl;
            break;
        case 5:
        case 4:
        case 3:
        case 2:
        case 1:
        case 0:
            cout<<"不及格"<<endl;
            break;
        default:
            cout<<"输入错误"<<endl;
            break;
    }

    return 0;
}
```

程序同8. A，将其改正确，即所有[0..100]之外的数据均给出“输入错误”即可

```
#include <iostream>
using namespace std;
int main()
{
    int score;
    cout << "请输入成绩[0-100]: ";
    cin >> score;

    if (score >= 0 && score <= 100) {
        switch (score / 10) {
            case 10:
            case 9:
                cout << "优" << endl;
                break;
            case 8:
                cout << "良" << endl;
                break;
            case 7:
                cout << "中" << endl;
                break;
            case 6:
                cout << "及格" << endl;
                break;
            case 5:
            case 4:
            case 3:
            case 2:
            case 1:
            case 0:
                cout << "不及格" << endl;
                break;
        }
    }
    else {
        cout << "输入错误" << endl;
    }

    return 0;
}
```



## §. 基础知识题 - 关系运算、逻辑运算与选择结构

### 8、switch-case语句

#### H. 思考

如果将成绩区间对应为：

[84-100]	- 优
[68-84)	- 良
[55-68)	- 及格
[0-55)	- 不及格

#### 1、用if-else语句完成该程序并贴图

```
计算.cpp  × homework3.c
homework
1  #include <iostream>
2  using namespace std;
3
4  int main()
5  {
6      int score;
7      cout << "请输入成绩[0-100]" << endl;
8      cin >> score;
9      if (score >= 0 && score <= 100) {
10         if (score >= 84 && score <= 100)
11             cout << "优" << endl;
12         else if (score >= 68 && score < 84)
13             cout << "良" << endl;
14         else if (score >= 55 && score < 68)
15             cout << "及格" << endl;
16         else if (score >= 0 && score < 55)
17             cout << "不及格" << endl;
18     }
19     else
20         cout << "输入错误" << endl;
21
22     return 0;
23 }
```

```
1  #include <iostream>
2  using namespace std;
3
4  int main()
5  {
6      int score;
7      cout << "请输入成绩[0-100]" << endl;
8      cin >> score;
9
10     switch (score) {
11         case 100:
12         case 99:
13         case 98:
14         case 97:
15         case 96:
16         case 95:
17         case 94:
18         case 93:
19         case 92:
20         case 91:
21         case 90:
22         case 89:
23         case 88:
24         case 87:
25         case 86:
26         case 85:
27         case 84:
28             cout << "优" << endl;
29             break;
30         case 83:
31         case 82:
32         case 81:
33         case 80:
34         case 79:
35         case 78:
36         case 77:
37         case 76:
38         case 75:
39         case 74:
40         case 73:
41         case 72:
42         case 71:
43         case 70:
44         case 69:
45         case 68:
46             cout << "良" << endl;
47             break;
48         case 67:
49         case 66:
50         case 65:
51         case 64:
52         case 63:
53         case 62:
54         case 61:
55         case 60:
56         case 59:
57         case 58:
58         case 57:
59         case 56:
60         case 55:
61             cout << "及格" << endl;
62             break;
63         default:
64             cout << "不及格" << endl;
65             break;
66     }
67
68     return 0;
69 }
```

#### 2、如果用switch语句，该如何实现？（如果程序太长，允许只截取能说明问题的部分即可）



## §. 基础知识题 - 关系运算、逻辑运算与选择结构

### 8、switch-case语句

#### H. 思考

如果将成绩区间对应为：

[84-100]	- 优
[68-84)	- 良
[55-68)	- 及格
[0-55)	- 不及格

3、如果学生成绩带小数点，即“xx.5”形式，能用if语句吗？能用switch语句吗？请解释原因

可以用if语句，不能用switch语句，应为Switch语句只能用int、char、short、long、bool等整型，不能用float、double等浮点型。

4、总结switch语句使用时的注意事项

Switch语句只能用int、char、short、long、bool等整型，不能用float、double等浮点型； case只能跟整型常量或常量表达式，不能跟变量，且不能存在两个相同的标签值；各语句序列最后一句应该为break，否则连续执行下一case语句，最后一个可省；对于超出范围的值可能无法给出准确结果。

5、switch-case语句能完全取代if-else吗？

不能，switch具有局限性。

需要文字描述的地方，尽可能用简单的文字描述清楚即可，不需要卷字数



## §. 基础知识题 - 关系运算、逻辑运算与选择结构

此页不要删除，也没有意义，仅仅为了分隔题目