

# § . 基础知识题 - 关系运算、逻辑运算与选择结构



要求:

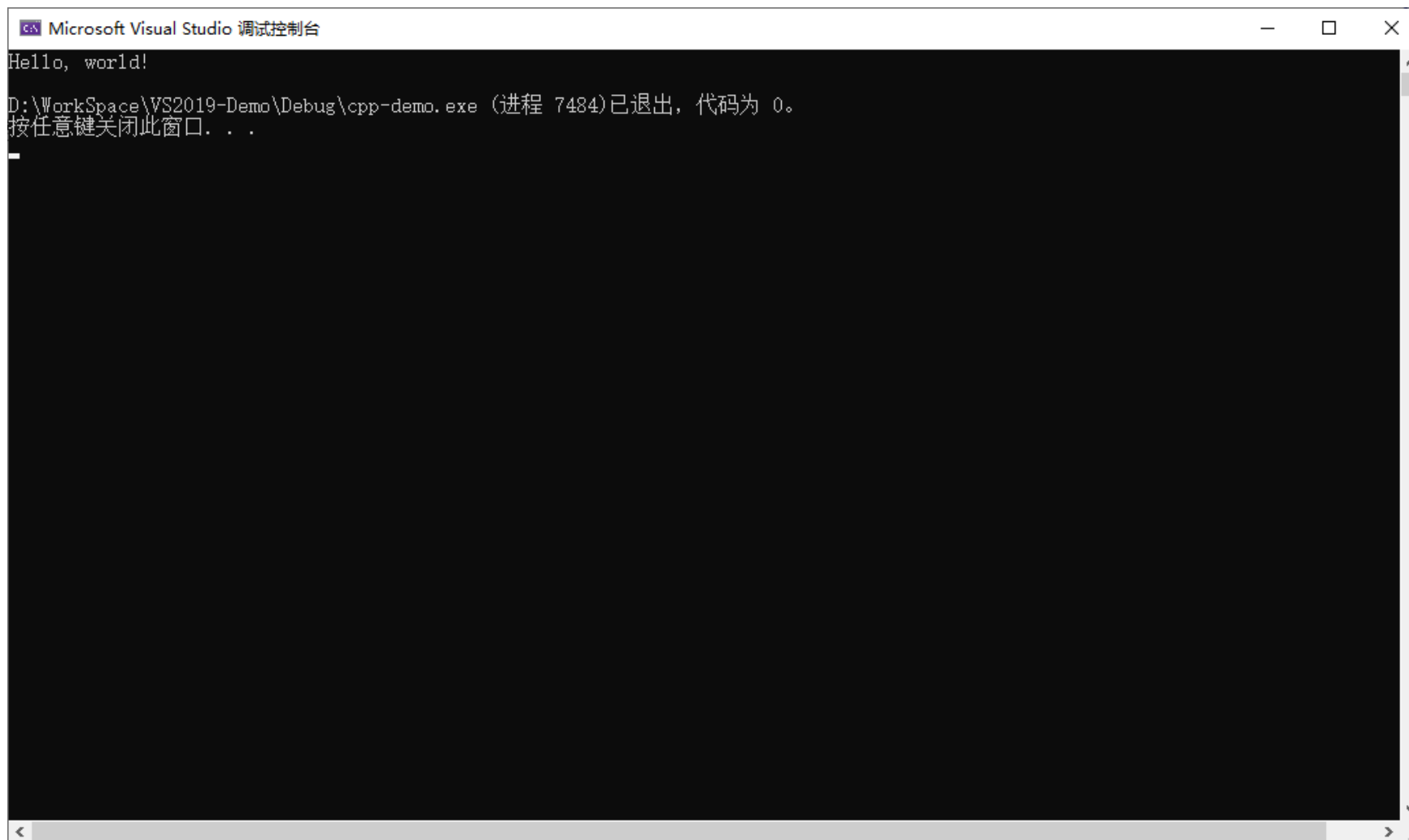
- 1、完成本文档中所有的题目并写出分析、运行结果
- 2、无特殊说明，均使用VS2022编译即可
- 3、直接在本文件上作答，**写出答案/截图（不允许手写、手写拍照截图）**即可；填写答案时，为适应所填内容或贴图，**允许调整**页面的字体大小、颜色、文本框的位置等
  - ★ 贴图要有效部分即可，不需要全部内容
  - ★ 在保证一页一题的前提下，具体页面布局可以自行发挥，简单易读即可
  - ★ **不允许**手写在纸上，再拍照贴图
  - ★ **允许**在各种软件工具上完成（不含手写），再截图贴图
  - ★ 如果某题要求VS+Dev的，则如果两个编译器运行结果一致，贴VS的一张图即可，如果不一致，则两个图都要贴
- 4、转换为pdf后提交
- 5、**3月26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

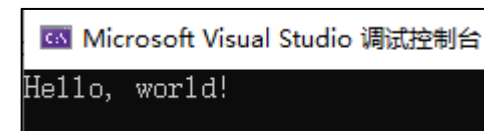


贴图要求：只需要截取输出窗口中的有效部分即可，如果全部截取/截取过大，则视为无效贴图

例：无效贴图

A screenshot of the Microsoft Visual Studio debug console window. The window is titled "Microsoft Visual Studio 调试控制台". It contains the text "Hello, world!" followed by a message indicating that the program "D:\Workspace\VS2019-Demo\Debug\cpp-demo.exe (进程 7484) 已退出, 代码为 0." and a prompt "按任意键关闭此窗口. . .". The window is large and shows the full context of the output.

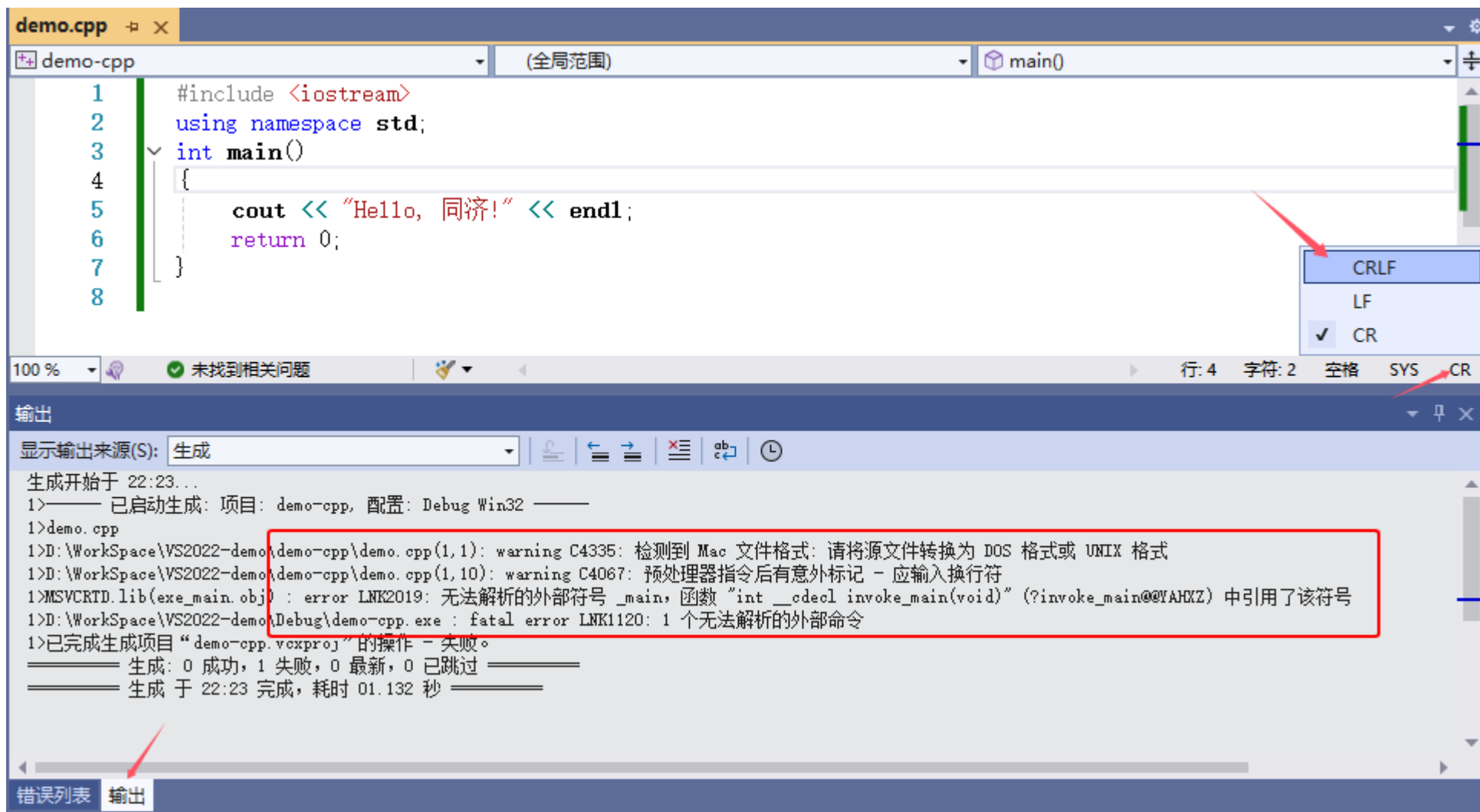
例：有效贴图

A screenshot of the Microsoft Visual Studio debug console window, cropped to show only the "Hello, world!" output. The window title is "Microsoft Visual Studio 调试控制台".



## §. 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;

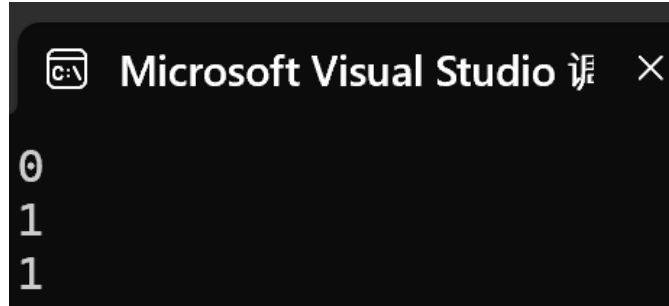
    d = a > b > c;
    cout << d << endl;

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

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

    return 0;
}
```

#### 1、贴运行结果



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

答：以第一个赋值表达式为例，根据运算符优先级顺序， $a > b$ 会先被计算，然后其返回值作为bool型变量，得出的0或1再参与后续的比大小运算，因此可能会导致结果不正确。



## §. 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>	<p>1、贴运行结果</p> <p>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 ? 答: <math>a &gt; b</math>返回值为1, <math>1 &gt; c</math>返回值为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 ? 答: <math>a &lt; b</math>返回值为0, <math>0 &lt; c</math>返回值为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 ? 答: <math>b &gt; a</math>返回值为0, <math>0 &lt; c</math>返回值为1 (文字简单说明即可)</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

### 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

```
0
1
1
4.76837e-08
0
1
```

```
0
1
1
4.76837e-08
0
1
```

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

```
0
1
1
4.76837e-08
0
1
```

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

[Error] 'fabs' was not declared in this scope; did you mean 'labs'?

#### 3、由本例得出的结论，实数进行相等比较时的通用方法是当两数相减的绝对值小于一个非常小的值时认为相等





## §. 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

```
0
1
0
1
1
1
```

```
0
1
0
1
1
1
```

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？

答：1e-6，1e-7不同。因为float的精度为六位有效数字，double的精度为15位有效数字，作为float存储时后面的精度会丢失，因此只有在double类型的两变量差值与1e-7进行比较时，两个值之间的差异才会体现。



## §. 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;

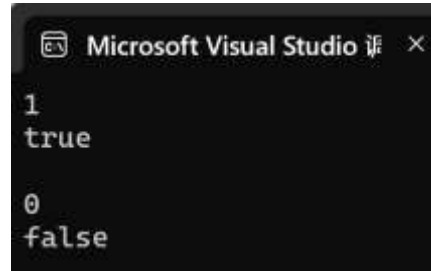
int main()
{
    cout << true    << endl;
    cout << "true"  << endl;

    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")

答: true是bool类型变量值, "true" 是字符串。

#### 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)

答: `cout<<boolalpha<<true<<endl;`  
`cout<<false<<endl;`



## §. 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;

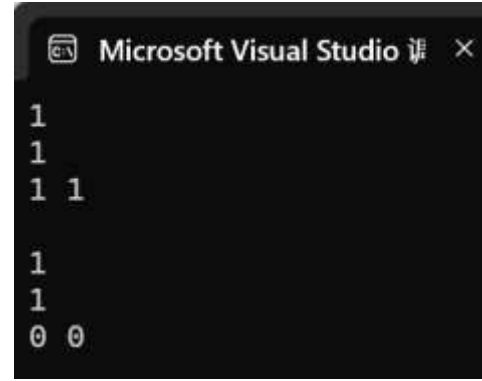
    cout << sizeof(true) << endl;
    cout << sizeof(k1) << endl;
    cout << k1 << ' ' << int(k1) << endl;

    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\_\_1或0\_\_

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)

答：bool型常量/变量在输出时会被转换为相应的整数值1(true)或0(false)。



# §. 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)

<pre>#include &lt;iostream&gt; using namespace std;  int main() {     bool k;      cin &gt;&gt; k;     cout &lt;&lt; k &lt;&lt; ' ' &lt;&lt; int(k) &lt;&lt; endl;      return 0; }</pre>	<div>1、输入0，输出是：</div> <div>2、输入1，输出是：</div> <div>3、输入123，输出是：</div> <div>4、输入true，输出是：</div> <div>5、输入false，输出是：</div> <div>总结bool型变量在输入时的规则： 答：对于整数，非零为true，零为false；cin无法读入字符串，输入错误。也就是需要输入代表true或者false的值。</div>	
---	--	--



## §. 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;

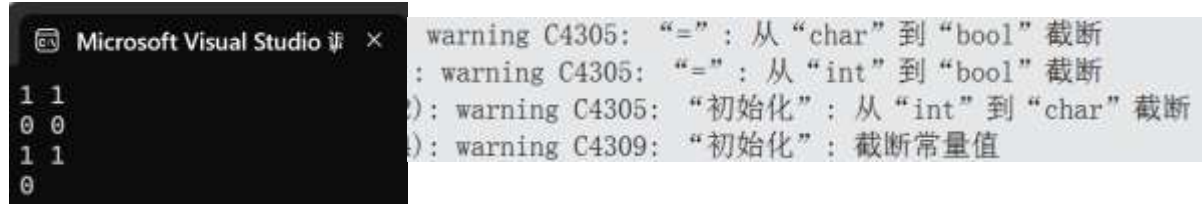
    k=0;
    cout << k << ' ' << (int)k << endl;

    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

答: bool为1个字节, int是4个, 将int类型值赋给bool会导致截断。Char能表示的值比bool更多, 也会导致warning。

#### 3、k='A' 是1字节赋值给1字节, 为什么还有warning?

答: 因为char可以表示0-255, bool只能表示0或1, 还是会导致数据丢失。

#### 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)

答: 应该是0000, 现在实际是1, 因为256是非零值, 表示true。

#### 5、为什么不 cout << c, 而是 (int)c ?

答: 确保输出的是整数值而不是字符(可能不可打印)。

#### 6、“非0为真0为假”这句话如何解释?

答: bool类型可以自动转换整数值, 只有0表示false(0), 其他都是true(1)



## §. 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)

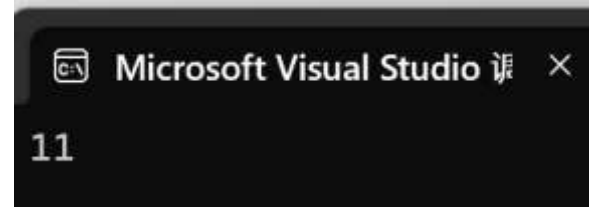
```
#include <iostream>
using namespace std;

int main()
{
    bool f=true;
    int a=10;

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

    return 0;
}
```

1、贴运行结果



2、当bool参与表达式计算时，当做\_一个值为1或0的整数\_



## §. 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

### 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



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

- ①  $a>b \Rightarrow 0$
- ②  $m=0$
- ③  $0\&\&(n=c>d) \Rightarrow 0$

3、短路运算的意思是：\_在与运算中，只要出现0，则整个表达式总为false，所以后面的表达式就不会被运算；在或运算中，只要出现1，整个表达式总为true，所以后面的表达式也不会被运算。\_\_\_\_\_



## §. 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;
//5 > 3 && 2 || 8 < 4 - !0, 1*2
int main()
{
    bool m = 0, n = 1;
    int p = 2, q = 3;

    cout << "m=" << m << " n=" << n << endl;
    (m = 5 > 3 && 2) || (n = 8 < 4 - !0);
    cout << "m=" << m << " n=" << n << endl;

    cout << "p=" << p << " q=" << q << endl;
    p = (5 > 3 && 2 || 8 < 4 - !0), (q =
1*2);
    cout << "p=" << p << " q=" << q << endl;

    return 0;
}
```

```
Microsoft Visual Studio 诶 ×
m=0 n=1
m=1 n=1
p=2 q=3
p=1 q=2
```

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;

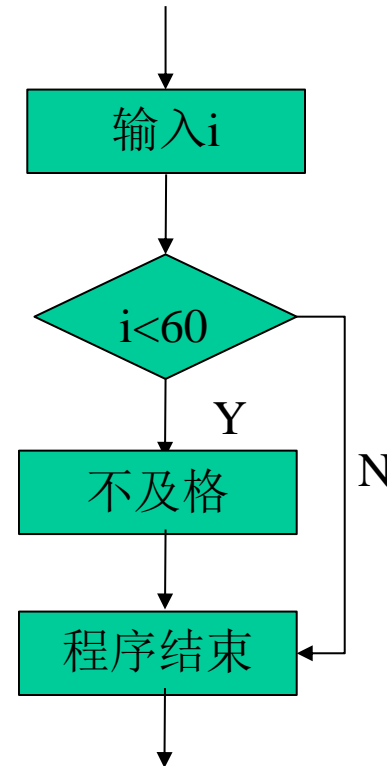
    if (i<60) {
        cout << "不及格" << endl;
    }
    cout << "程序结束" << endl;

    return 0;
}
```

```
Microsoft Visual Studio 调 ×
请输入成绩[0-100]
34
不及格
程序结束
```

```
Microsoft Visual Studio 调 ×
请输入成绩[0-100]
74
程序结束
```

- 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;

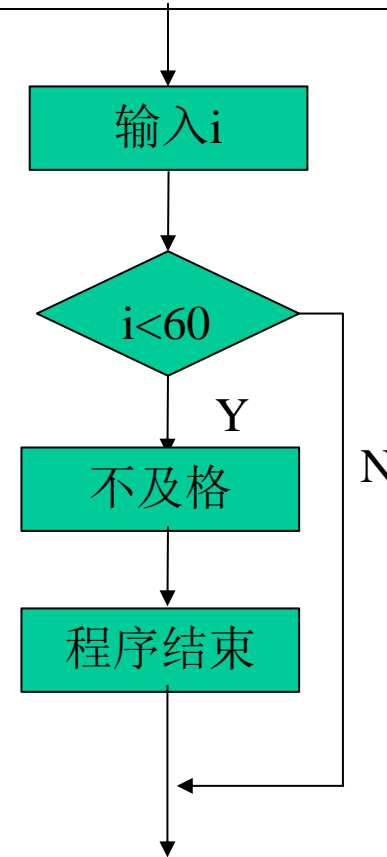
    cout<<"请输入成绩[0-100]"<<endl;
    cin >> i;

    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)

<pre>#include &lt;iostream&gt; using namespace std;  int main() {     int i;      cout&lt;&lt;"请输入成绩[0-100]"&lt;&lt;endl;     cin &gt;&gt; i;      if (i&lt;60;) {         cout &lt;&lt; "不及格" &lt;&lt; endl;         cout &lt;&lt; "程序结束" &lt;&lt; endl; //未缩进     }      return 0; }</pre>	<div>贴编译错误并给出解释</div> <div>warning C4552: " &lt; ": 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)</div> <div>答: if() 里的应该是一个表达式, 不能是一个语句。</div>
--	--



## §. 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?

答: 不能, 否则输入(89, 90)间的小数时会输出输入错误。

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

答: 虽然逻辑不太正确( $i=90$ 时满足两个分支条件), 但是运行正确。因为已经进入了第一个if分支, 就不会再进入下一个else if分支。





## §. 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 答：自上而下，忽略{，以}为基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

```
int i;
cout << "请输入成绩[0-100]" << endl;
cin >> i;
if (i > 60)
{
    if (i >= 80)
    {
        cout << "优良" << endl;
    }
    else
    {
        cout << "中差" << endl;
    }
}
cout << "啊?" << endl;
else
{
    if (i = 60)
        cout << "合格" << endl;
    else
        cout << "不合格" << endl;
}

return 0;
}
```

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

error C2181: 没有匹配 if 的非法 else

2、请说明错误原因（如果计算机的报错和你的认知不同，想明白为什么）  
答：编译器会认为从上到下看，是单if、并列的cout，因此else非法



# §. 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 1 3\_\_





# §. 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>答：表达式2和表达式3必须为同类型、同语言（相容），否则无法确定条件表达式的值类型。</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)  
(100, 110)和(-10, 0)





# §. 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

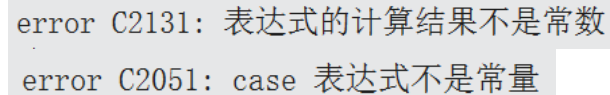
C. 观察下列程序的运行结果，回答问题并将程序的运行结果截图贴上(如果有错则贴错误信息截图)

```
#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

答：k+2不是一个常量。case后面必须是常量表达式，不能是变量或非常量表达式。



## §. 基础知识题 - 关系运算、逻辑运算与选择结构

### 8、switch-case语句

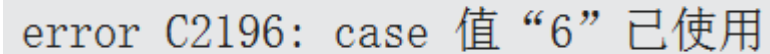
D. 观察下列程序的运行结果，回答问题并将程序的运行结果截图贴上(如果有错则贴错误信息截图)

```
#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



2、解释错误原因

答：case后面的常量重复了。



## §. 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)

```
#include <iostream>
using namespace std;
int main()
{
    float 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;
}
```

在8.A的基础上，将score从int改为float

1、给出编译错误的截图

error C2450: 类型为“float”的 switch 表达式无效

2、解释错误原因

答：switch()里应该是整型，这里是float型。



# §. 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



2、解释break的作用

答：加break之后，执行完相应case之后就不会连续执行下一个语句





## §. 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;
    if (score <= 100 && score >= 0) {
        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;
        }
    }
    else
        cout << "输入错误" << endl;
    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



## §. 基础知识题 - 关系运算、逻辑运算与选

### 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

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）

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

答：能用if（因为可以用范围进行语句分支），switch不行（只能处理整型，匹配单个值，除非枚举或者转换为整数，但是很麻烦）

4、总结switch语句使用时的注意事项

答：if()中只能是整数，switch后只能是整型常量，case最后要加break

5、switch-case语句能完全取代if-else吗？

答：不能，原因与3一样，在处理小数相关问题时麻烦且可能不精确。

```
#include <iostream>
using namespace std;
int main()
{
    cout << "输入在[0, 100]内的数字" << endl;
    int k;
    cin >> k;
    if (k <= 100 && k >= 0)
    {
        if (k <= 100 && k >= 84)
            cout << "优" << endl;
        if (k < 84 && k >= 68)
            cout << "良" << endl;
        if (k < 68 && k >= 55)
            cout << "中" << endl;
        else
            cout << "不及格" << endl;
    }
    else
        cout << "输入错误" << endl;
    return 0;
}

if (k <= 100 && k >= 0)
{
    if (k <= 100 && k >= 84)
        t = 5;
    if (k < 84 && k >= 68)
        t = 4;
    if (k < 68 && k >= 55)
        t = 3;
    if (k < 55)
        t = 2;
    switch (t)
    {
        case 5:
            cout << "优" << endl;
            break;
        case 4:
            cout << "良" << endl;
            break;
        case 3:
            cout << "及格" << endl;
            break;
        default:
            cout << "不及格" << endl;
    }
}
```



## §. 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