

§. 基础知识题 - 关系运算、逻辑运算与选择结构



要求:

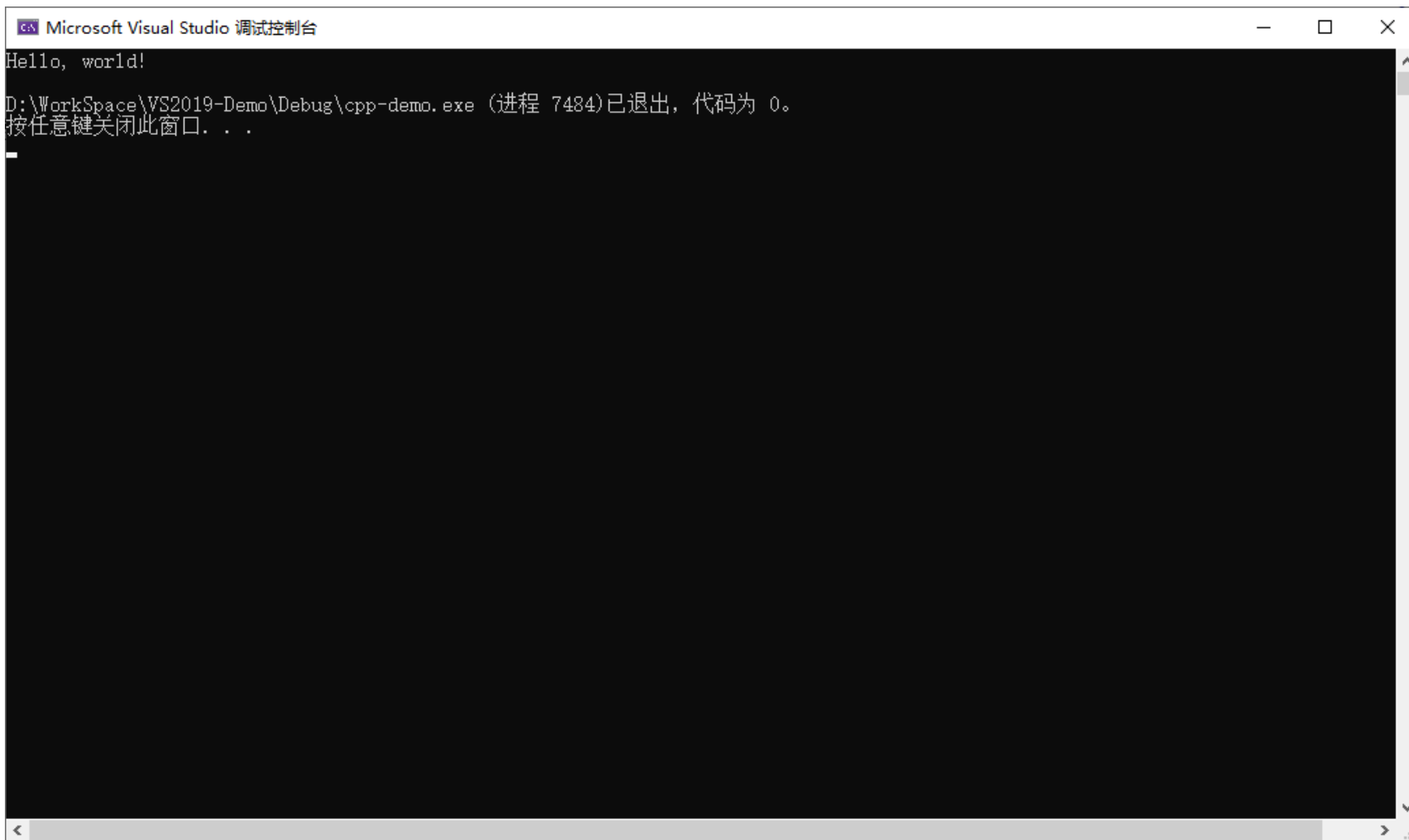
- 1、完成本文档中所有的题目并写出分析、运行结果
- 2、无特殊说明，均使用VS2022编译即可
- 3、直接在本文件上作答，**写出答案/截图（不允许手写、手写拍照截图）**即可；填写答案时，为适应所填内容或贴图，**允许调整**页面的字体大小、颜色、文本框的位置等
 - ★ 贴图要有效部分即可，不需要全部内容
 - ★ 在保证一页一题的前提下，具体页面布局可以自行发挥，简单易读即可
 - ★ **不允许**手写在纸上，再拍照贴图
 - ★ **允许**在各种软件工具上完成（不含手写），再截图贴图
 - ★ 如果某题要求VS+Dev的，则如果两个编译器运行结果一致，贴VS的一张图即可，如果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



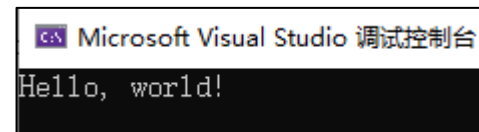
§. 基础知识题 - 关系运算、逻辑运算与选择结构

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



例：有效贴图

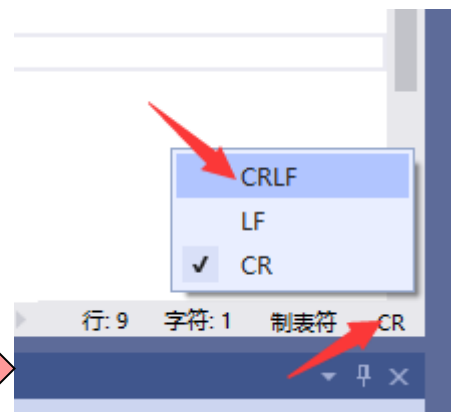




§. 基础知识题 - 关系运算、逻辑运算与选择结构

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;

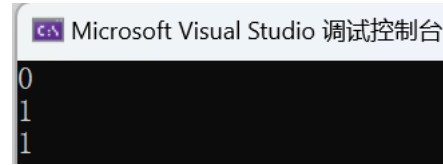
    d = a > b > c;
    cout << d << endl;

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

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

    return 0;
}
```

1、贴运行结果



2、VS下为什么会有三个warning? 说说你的理解

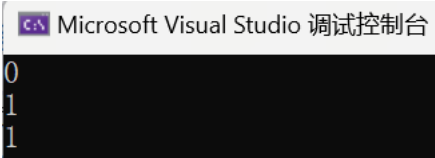
此程序试图将后面比较表达式的正确与否的bool值赋给d，但bool类型与int类型实际上是不同的，因此编译器报warning



§. 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 <iostream> using namespace std; int main() { int a=3, b=2, c=1, d; d = a > b > c; cout << d << endl; d = a < b < c; cout << d << endl; d = b > a < c; cout << d << endl; return 0; }</pre>	<div>1、贴运行结果</div> <div></div> <div>2、$a > b > c$这个式子，按人的常规理解($3 > 2$且$2 > 1$)是正确的，为什么结果是0 ? $a < b < c$这个式子，按人的常规理解($3 < 2$且$2 < 1$)是错误的，为什么结果是1 ? $b > a < c$这个式子，按人的常规理解($2 > 3$且$3 < 1$)是错误的，为什么结果是1 ? (文字简单说明即可) 对$a > b > c$, 计算机先比较$a > b$，此表达式正确，该表达式整体的值即为1再用c的值；与1比较，表达式错误，即为0，输出0； 对$a < b < c$, 计算机先比较$a < b$，不对，整个式子的值为0，但$0 < c$正确，故最后输出1； 同理，对于$b > a < c$, 计算机先比较$b > a$，错误，值为0，然后进行$0 < c$，正确，输出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)

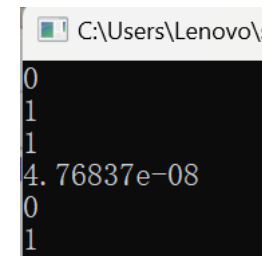
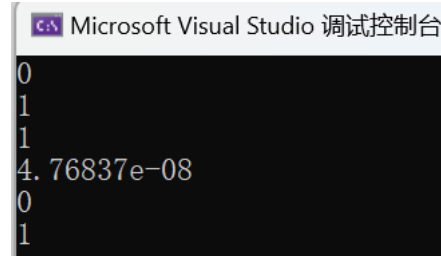
```
#include <iostream>
#include <cmath>
using namespace std;

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

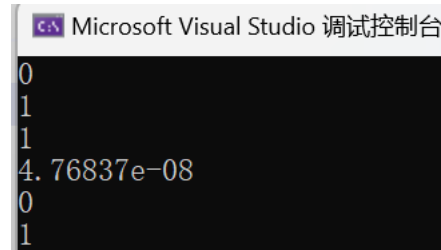
    float f2 = 1.2;
    cout << (f2 - 1.2) << endl;
    cout << (f2 == 1.2) << endl;
    cout << (fabs(f2-1.2) < 1e-6) << endl;

    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



[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 使用“=”输出bool值



§. 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)

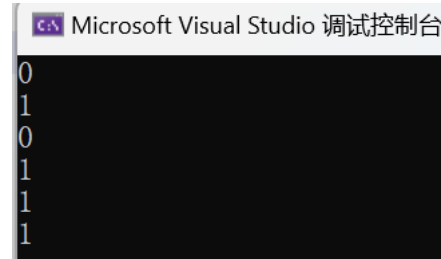
```
#include <iostream>
#include <cmath> //VS可不加
using namespace std;

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



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 $\text{fabs}(**)<1e-6$ 和 $\text{fabs}(**)<1e-7$ 在float和double下的表现，哪个相同？哪个不同？为什么？
1e-6相同，1e-7不同，这与数据类型的精确范围与精确位数有关



§. 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

Microsoft Visual Studio 调试控制台

```
1
true
0
false
```

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

true被编译器认为是一个bool类型变量，输出1； " true" 则是一个字符串，直接输出；类似有false和 "false" 。

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)
在cout语句中加boolalpha格式控制符



§. 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0/1

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)
默认true为1，false为0输出



§. 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)

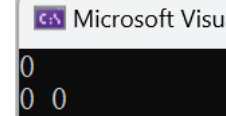
```
#include <iostream>
using namespace std;

int main()
{
    bool k;

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

    return 0;
}
```

1、输入0，输出是：



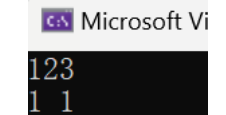
```
Microsoft Visual Studio
0
0 0
```

2、输入1，输出是：



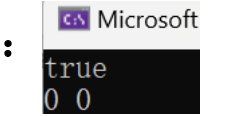
```
Microsoft Visual Studio
1
1 1
```

3、输入123，输出是：



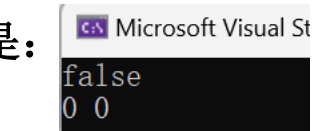
```
Microsoft Visual Studio
123
1 1
```

4、输入true，输出是：



```
Microsoft Visual Studio
true
0 0
```

5、输入false，输出是：



```
Microsoft Visual Studio
false
0 0
```

总结bool型变量在输入时的规则：0为false，非0均为true，字符串等非0输入均为false，即为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

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

```
Microsoft
1 1 cpp(7,9): warning C4305: "=": 从"char"到"bool"截断
0 0 cpp(13,9): warning C4305: "=": 从"int"到"bool"截断
1 1 cpp(16,12): warning C4305: "初始化": 从"int"到"char"截断
0 cpp(16,14): warning C4309: "初始化": 截断常量值
```

2、解释VS下warning的意思

在不同类型的相互赋值间可能出现误差，如本程序试图将char型赋给bool型

3、k='A' 是1字节赋值给1字节，为什么还有warning?

bool型变量比较特殊，值为0/1

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)

0，现在实际是1，编译器认为原本的数非0即全为true，即为1

5、为什么不 cout << c，而是 (int)c ?

cout << c无打印输出，256赋给char型会发生溢出，导致值为0，而输出char型时是输出ASCII码对应的字符，对应的是空字符。

6、“非0为真0为假”这句话如何解释?

对于bool型变量，输入若为非0，编译器认为true，输出1；若为0，认为false，输出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)

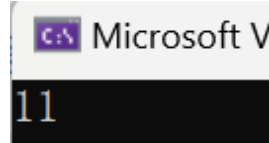
```
#include <iostream>
using namespace std;

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

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

    return 0;
}
```

1、贴运行结果



2、当bool参与表达式计算时，当做int型



§. 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	假	假	真	真
1	0	假	真	假	真
0	1	真	假	假	真
0	0	真	真	假	假

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)

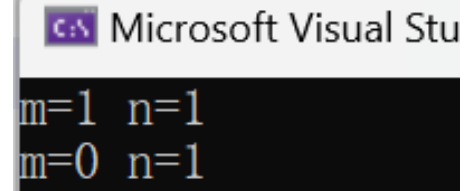
```
#include <iostream>
using namespace std;

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 Stu

```
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)

1: $a > b \Rightarrow 0$

2: $m = 0$

3: n 不再求解, 保持原值1

3、短路运算的意思是: 仅当必须执行下一个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;

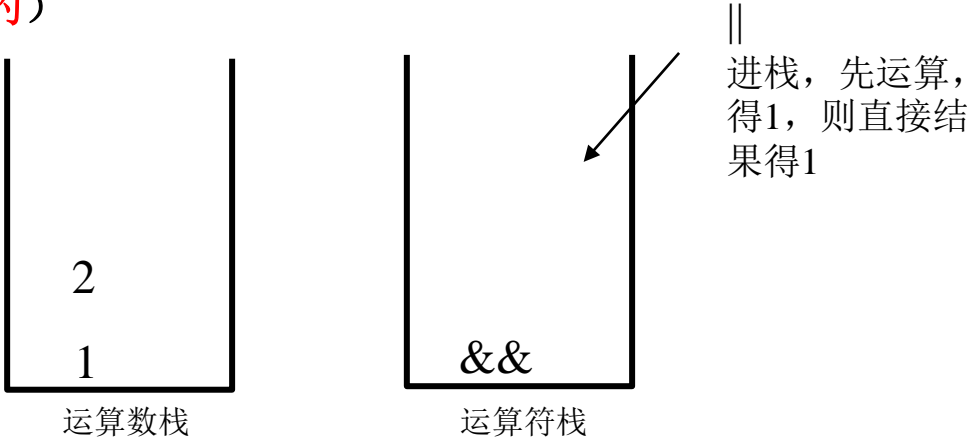
int main()
{
    int a=0,b=0;
    cout << "a=" << a << "b=" << b << endl;
    5 > 3 && 2 || 8 < 4 - (a!=0), 5 < 3 && 2
    || 8 < 4 - (b!=0);
    cout << "a=" << a << "b=" << b << endl;

    return 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||后的运算符。（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;
}
```

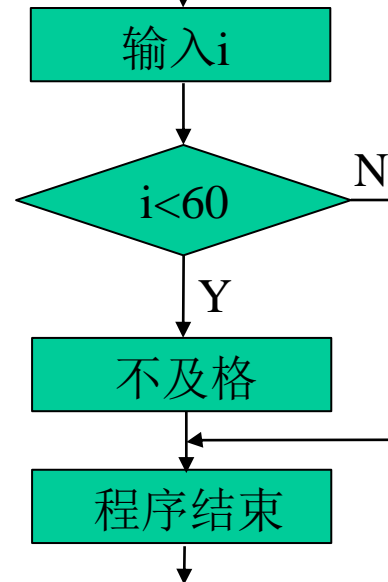
1、输入34，贴运行结果

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

2、输入74，贴运行结果

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

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;
}
```

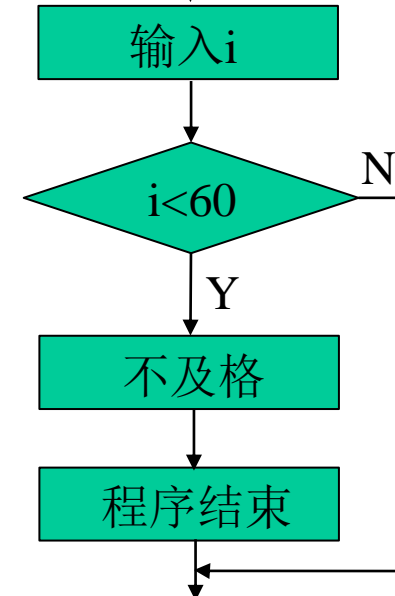
1、输入34，贴运行结果

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

2、输入74，贴运行结果

```
Microsoft Visual Studio 调试控制台
请输入成绩[0-100]
74
```

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

```
cpp(11,11): warning C4552: " < ": 未使用表达式结果
cpp(11,15): error C2429: 语言功能 "if/switch 中的 init-statement" 需要编译器标志 "/std:c++17"
cpp(11,16): error C2059: 语法错误: ")"
cpp(11,18): error C2143: 语法错误: 缺少 ";" (在 "{" 的前面)
```

If语句的语法错误，表达式后无;



§. 基础知识题 - 关系运算、逻辑运算与选择结构

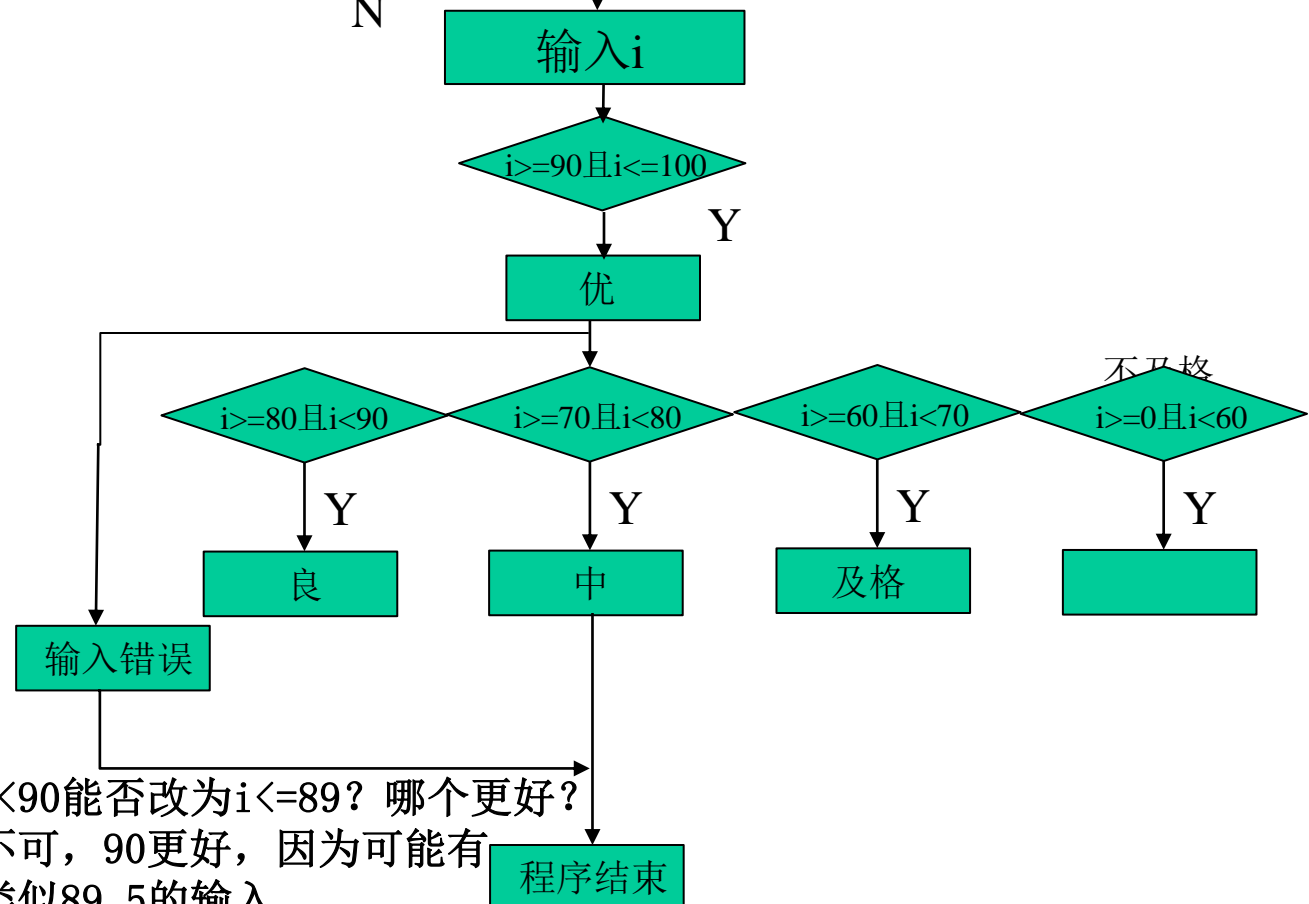
5、if语句 - 基本使用

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?

不可, 90更好, 因为可能有类似89.5的输入

3、 $i < 90$ 能否改为 $i \leq 90$? 运行是否正确?

可, 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

error C2181: 没有匹配 if 的非法 else

2、请说明错误原因

选择结构if多支语句中不能插入其他的语句

```
#include <iostream>  
using namespace std;  
  
int main()  
{  
    int a=2,b=5;  
    if (a>1) {  
        if (a<4) {  
            cout << "a=" << a << endl;  
        }  
        else {  
            cout << "wrong(a)" << endl;  
        }  
        cout << "-----" << endl;  
    }  
    else {  
        if (b>3) {  
            cout << "b=" << b << endl;  
        }  
        cout << "wrong(b)" << endl;  
    }  
    return 0;  
}
```



§ . 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;

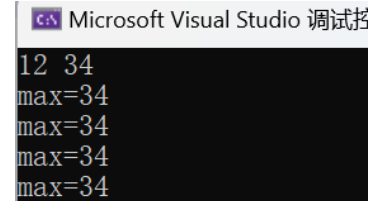
    a > b ? cout << "max=" << a << endl : cout << "max=" << b << endl; //1

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

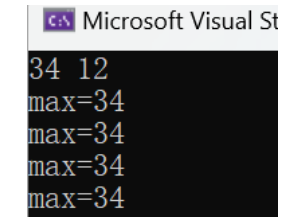
    printf("max=%d", a>b?a:b); //3

    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 <iostream> using namespace std; int main() { int a=1, b=2; a==1 ? "Hello" : 123; a>b ? cout << a : printf("%d", b); a==1 ? 'A' : 123; return 0; }</pre>	<div>1、给出编译报错的截图</div> <div>error C2446: “:” : 没有从 “int” 到 “const char [6]” 的转换</div> <div>error C2678: 二进制 “?” : 没有找到接受 “std::basic_ostream<char, std::char_traits<char>>” 类型的左操作数的运算符(或没有可接受</div> <div>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) 表达式1、2、3的类型可以不同，但表达式2、3的类型必须相容。</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

8、switch-case语句

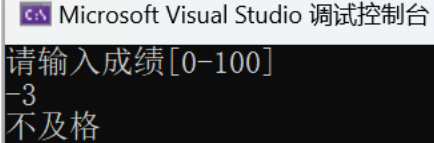
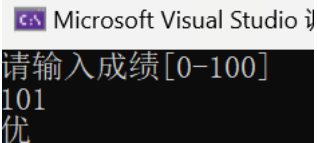
A. 观察下列程序的运行结果，回答问题并将程序的运行结果截图贴上(如果有错则贴错误信息截图)

```
#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)
[-9, -1]和[101, 109]





§. 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)

<pre>#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; }</pre>	<p>在8. A的基础上</p> <ol style="list-style-type: none">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 <p>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)</p> <p>结论：8. A和8. B <u>完全一致</u> (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</p>
--	--



§ . 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)

<pre>#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; }</pre>	<p>在8.B的基础上，将k从const int改为int</p> <p>1、给出编译错误的截图</p> <pre>cpp(20, 12): error C2131: 表达式的计算结果不是常数 cpp(20, 10): cpp(20, 10): cpp(20, 5): error C2051: case 表达式不是常量</pre> <p>2、解释错误原因</p> <p>int定义的是一个整型的变量，而const int定义的是一个整型的常量，而case表达式不能是变量表达式。</p>
--	--



§. 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

各整型常量表达式的值应各不相同。



§ . 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 <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; }</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</p> <p>error C2450: 类型为“float”的 switch 表达式无效</p> <p>2、解释错误原因</p> <p>要求表达式的最终取值为整型。</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)

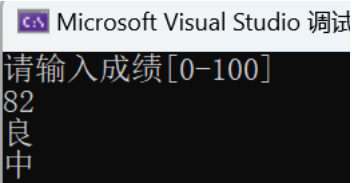
```
#include <iostream>
using namespace std;
int main()
{
    int score;
    cout<<"请输入成绩[0-100]"<<endl;
    cin >> score;
    switch(score/10) {
        case 10:
        case 9:
            cout<<"优"<<endl;
            break;
        case 8:
            cout<<"良"<<endl;

        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



2、解释break的作用

跳出case语句的执行，若无会连续执行下一case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;
    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

```
1  #include <iostream>
2  using namespace std;
3  int main()
4  {
5      int score;
6      cout << "请输入成绩[0-100]" << endl;
7      cin >> score;
8      if (score >= 0 && score <= 100) {
9          switch (score / 10) {
10             case 10:
11             case 9:
12                 cout << "优" << endl;
13                 break;
14             case 8:
15                 cout << "良" << endl;
16                 break;
17             case 7:
18                 cout << "中" << endl;
19                 break;
20             case 6:
21                 cout << "及格" << endl;
22                 break;
23             default:
24                 cout << "不及格" << endl;
25                 break;
26             }
27         }
28         else
29             cout << "输入错误" << endl;
30         cout << "程序结束" << endl;
31         return 0;
32     }
```

C:\> Microsoft Visual Studio 诶
请输入成绩[0-100]
104
输入错误
程序结束

C:\> Microsoft Visual Studio
请输入成绩[0-100]
-3
输入错误
程序结束



§. 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

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

int main()
{
    int i;
    cout << "请输入成绩[0-100]" << endl;
    cin >> i;
    if (i >= 84 && i <= 100)
        cout << "优" << endl;
    else if (i >= 68 && i < 84)
        cout << "良" << endl;
    else if (i >= 55 && i < 68)
        cout << "及格" << endl;
    else if (i >= 0 && i < 55)
        cout << "不及格" << endl;
    else
        cout << "输入错误" << endl;
    cout << "程序结束" << endl;
    return 0;
}
```

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

可用if，不能用switch，switch要求是整型。

4、总结switch语句使用时的注意事项

整型；表达式值不同；语句序列最后break。

5、switch-case语句能完全取代if-else吗？

不能，if-else适用于大部分场景，switch-case语句只在某些特定时候更简便。

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

```
if (score >= 0 && score <= 100) {
    switch (score / 14) {

        case 7:
        case 6:

            cout << "优" << endl;
            break;

    }

    switch (score / 17) {
        case 4:
            cout << "良" << endl;
            break;
    }

    switch (score / 11) {
        case 5:
            cout << "及格" << endl;
            break;
    }

    switch (score / 11) {
        case 4:
        case 3:
        case 2:
        case 1:
        case 0:
            cout << "不及格" << endl;
            break;
    }

} else
    cout << "输入错误" << 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