

§ . 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、**9月29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

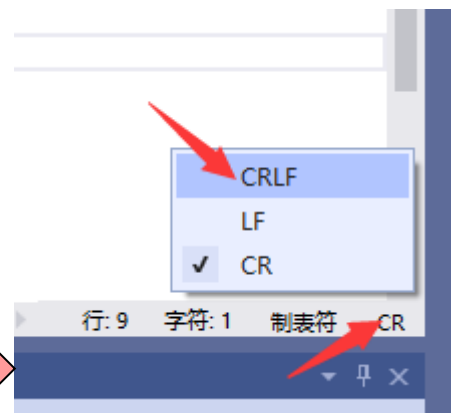
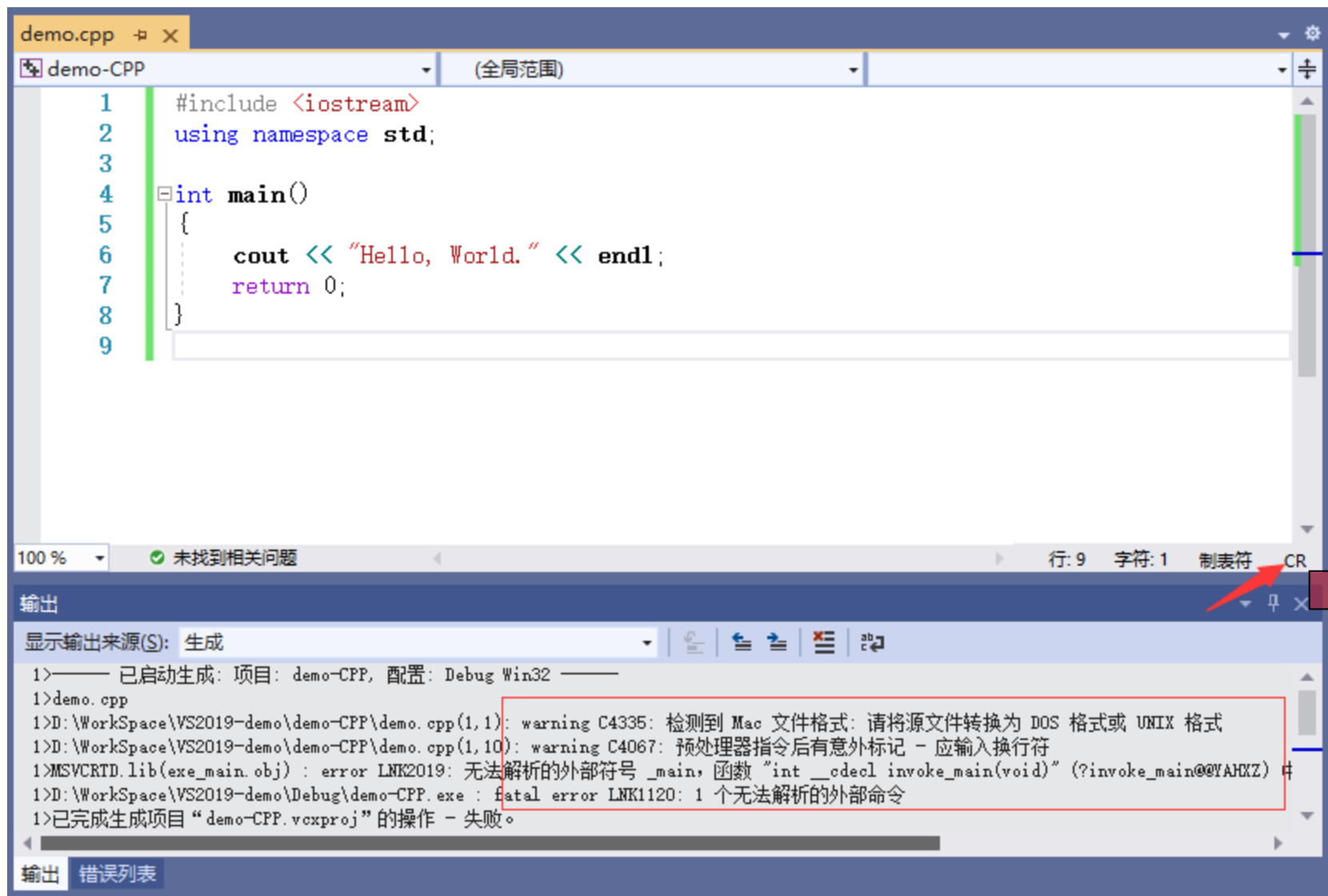
```
Microsoft Visual Studio 调试控制台  
Hello, world!
```



§. 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

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;

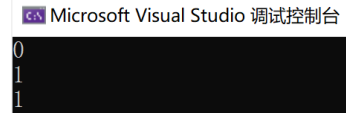
    d = a > b > c;
    cout << d << endl;

    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

关系运算符返回的是bool类型，只能返回0或1，使用bool继续进行关系运算比较，有隐患



§. 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)

```
#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;
}
```

1、贴运行结果

Microsoft Visual Studio 调试控制台



2、 $a > b > c$ 这个式子，按常规理解， $3 > 2 > 1$ 是正确的，为什么结果是0？

$3 > 2$ return 1, $1 > 1$ return 0

$a < b < c$ 这个式子，按常规理解， $3 < 2 < 1$ 是错误的，为什么结果是1？

$3 < 2$ return 0, $0 < 1$ return 1

$b > a < c$ 这个式子，按常规理解， $2 > 3 < 1$ 是错误的，为什么结果是1？

$2 > 3$ return 0, $0 < 1$ return 1



§. 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 b = 1.1;
    cout << (b - 1.1) << endl;
    cout << (b == 1.1) << endl;
    cout << (fabs(b - 1.1) < 1e-6) << endl;

    float c = 1.0;
    cout << (c - 1.0) << endl;
    cout << (c == 1.0) << endl;
    cout << (fabs(c - 1.0) < 1e-6) << endl;

    return 0;
}
```

1、贴VS+Dev下的运行结果

```
2.38419e-08
0
1
0
1
1
1

2.38419e-08
0
1
0
1
1
1
```

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

```
2.38419e-08
0
1
0
1
1
1
```



```
D:\同济高程\homework\hello world\helloworld.cpp In function 'int main()':
9 14 D:\同济高程\homework\hello world\helloworld.cpp [Error] 'fabs' was not declared in this scope; did you mean 'la...
```

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fabs判断两数之差，在合理范围内即可说明相等



§. 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



```
1>D:\同济\高程\homework\hello world\helloworld.cpp(13,32): warning C4305: "初始化": 从 "double" 到 "float" 截断
1>D:\同济\高程\homework\hello world\helloworld.cpp(14,32): 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？

Float 相同，double 不同。

Float 是六位有效数字，只能记录 $1e-6$ 及以上的误差

Double 是十五位有效数字， $1e-6$ $1e-7$ 均能体现误差



§. 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为bool类型，输出为1

"true" 为字符串

3、进阶思考：如果想使true和false在屏幕上输出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

在输出前先加一句 cout<<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;

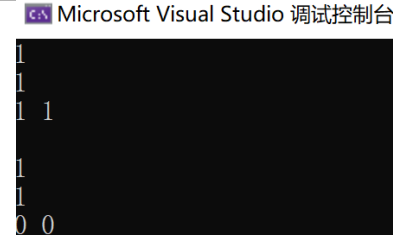
    cout << sizeof(true) << endl;
    cout << sizeof(k1) << endl;
    cout << k1 << ' ' << int(k1) << endl;

    cout << endl;

    bool k2 = false;
    cout << sizeof(false) << endl;
    cout << sizeof(k1) << 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)

常量变量true输出1，false输出0，转换为int得到的结果相同



§. 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)

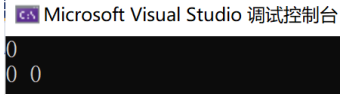
```
#include <iostream>
using namespace std;

int main()
{
    bool k;

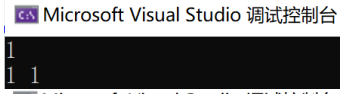
    cin >> k;
    cout << k << ' ' << int(k) << endl;

    return 0;
}
```

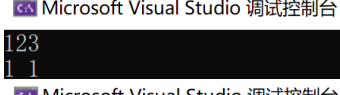
1、输入0，输出是：



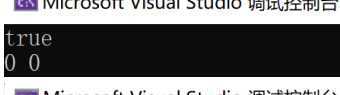
2、输入1，输出是：



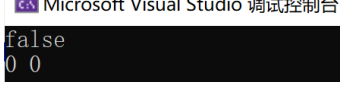
3、输入123，输出是：



4、输入true，输出是：



5、输入false，输出是：



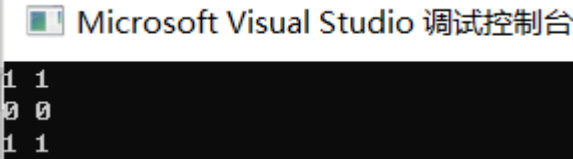
总结bool型变量在输入时的规则：
输入非零，值为1，输入0，值为0，输入字符串，值为0



§. 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)

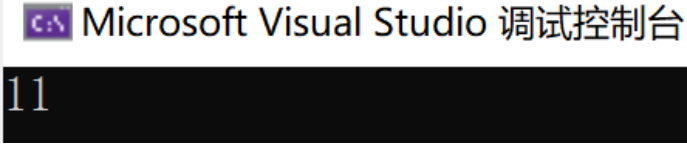
<pre>#include <iostream> using namespace std; int main() { bool k; k='A'; cout << k << ' ' << (int)k << endl; k=0; cout << k << ' ' << (int)k << endl; k=256; cout << k << ' ' << (int)k << endl; return 0; }</pre>	<div>1、贴运行结果</div> <div></div> <div>2、解释VS下waring的意思 ‘A’ 256转化为bool类型，发生隐藏的类型转换</div> <div>3、k='A' 是1字节赋值给1字节，为什么还有warning? Bool的值只有0与1，而char还有其他值</div> <div>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？ K应该为00000000 其他类型转化为bool类型，并不是强行高位截断</div> <div>5、“非0为真0为假”这句话如何解释？ 当其他类型转化为bool类型，如果值不等于0，转为true赋值给bool，如果值等于0，转为false赋值给bool。</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

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() { bool f=true; int a=10; a=a+f; cout << a << endl; return 0; }</pre>	<p>1、贴运行结果</p>  <p>2、当bool参与表达式计算时，当做__0或1__</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

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)

$a>b$ return 0, $m=0$

$\&\&$ 短路, 不进行后续运算

3、短路运算的意思是: _____ 当运算到达某处表达式的值能够确定, 便不进行后续运算 _____



§. 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;
```

```
int main()
{
    return 0;
}
```

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

```
int main()
{
    int m = 2;
    5 > 3 && 2 || (m=(8 < 4)) - !0;
    cout << m << endl;
    return 0;
}
```

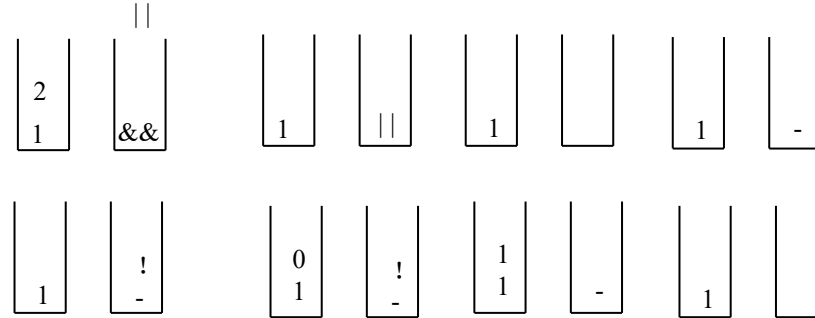


有以下逻辑表达式

$5 > 3 \ \&\& \ 2 \ || \ 8 < 4 - !0$

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 $8 < 4 - !0$ 存在/不存在短路运算存在

2、用栈方式画出从 $||$ 进栈开始（即第一张图为 $||$ 在栈顶），到整个表达式求值完成的过程（允许加页）





§. 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 调试控制台

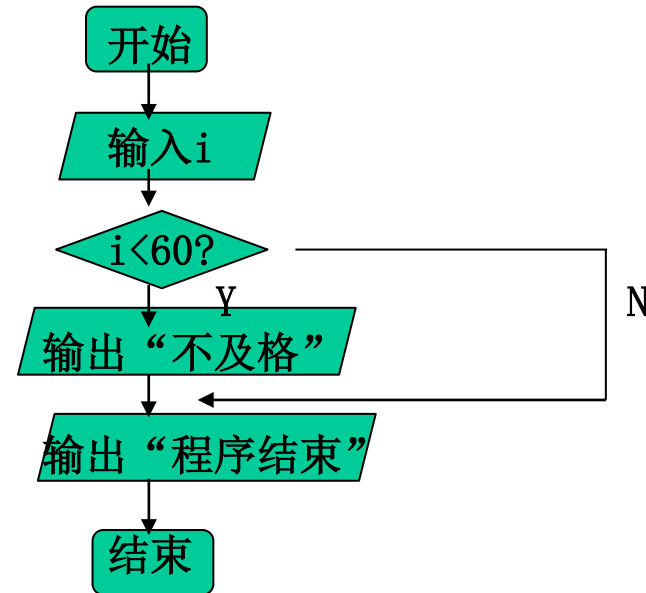
```
请输入成绩[0-100]
34
不及格
程序结束
```

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
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



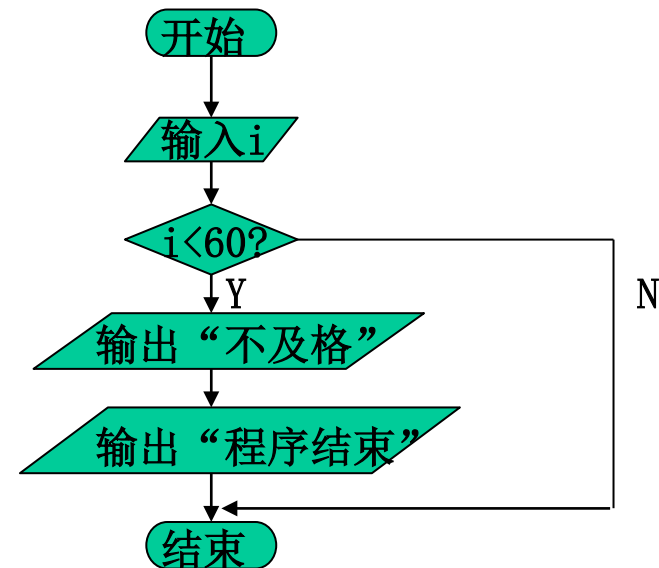
```
Microsoft Visual Studio 调试控制台
请输入成绩[0-100]
34
不及格
程序结束
```

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

✘ C2429 语言功能 "if/switch 中的 init-statement" 需要编译器标志 "/std:c++17"
✘ C2059 语法错误:)"
✘ C2143 语法错误: 缺少";"(在"{"的前面)
abc E0018 应输入")"

If括号内应为表达式，而不是语句



§. 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>=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;
}
```

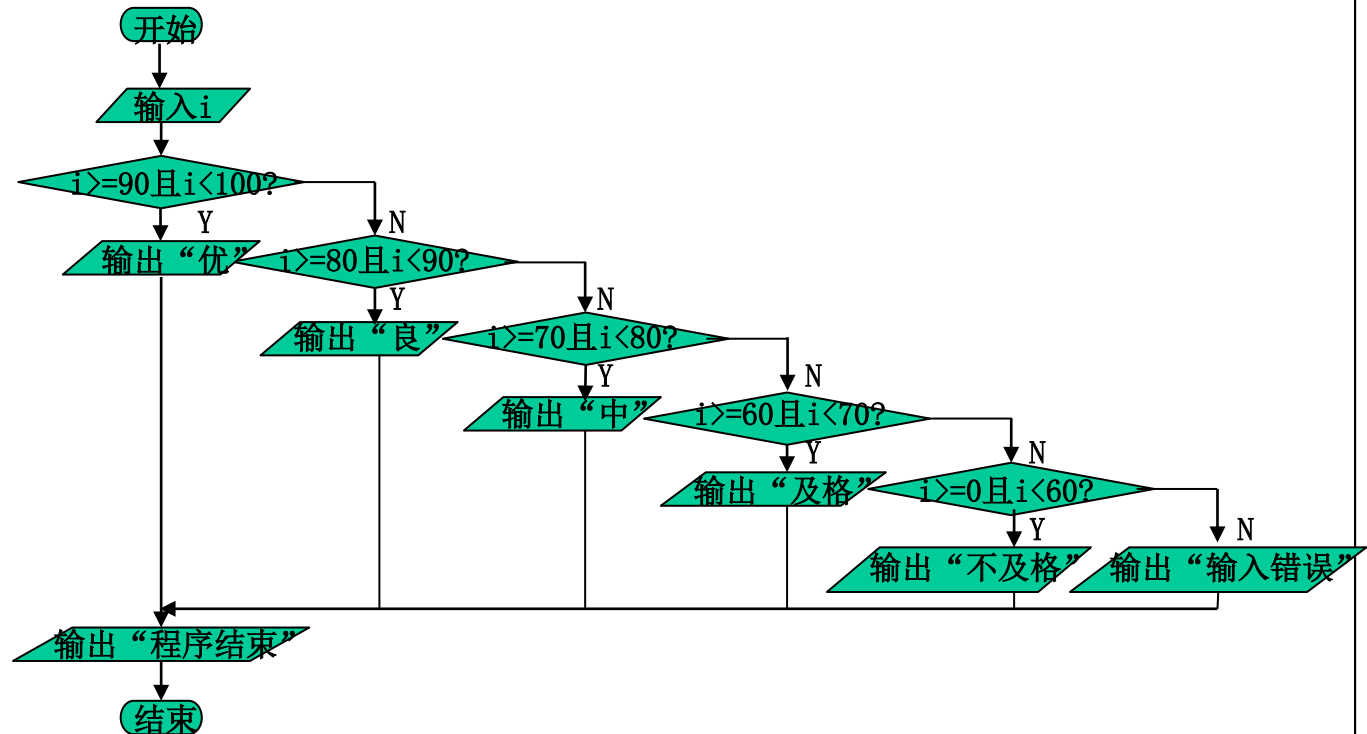
1、给出程序的流程框图(注意字体的清晰可辨)

2、 $i < 90$ 能否改为 $i \leq 89$? 哪个更好?

能改，但 $<$ 更好，当 i 变为实型时， \leq 无法考虑89-90之间的成绩

3、 $i < 90$ 能否改为 $i \leq 90$? 运行是否正确?

不能，无法考虑90的情况。当输入90时，运行错误





§. 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 }与向上最近未配对的{配对
---	--



§. 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，并给出该程序的编译错误截图

2、请说明错误原因

在if else之间插入语句，编译器认为else单独列出，没有与if匹配，报错

```
#include <iostream>  
using namespace std;  
  
int main()  
{  
    int i;  
    cin >> i;  
    if (i > 0) {  
        if (i > 1) {  
            cout << 1 << endl;  
        }  
        else  
            cout << 2 << endl;  
        cout << 3 << endl;  
    }  
    cout << 4 << endl;  
    else {  
        if (i > 2)  
            cout << 5 << endl;  
        cout << 6 << endl;  
    }  
    return 0;  
}
```

✖ C2181 没有匹配 if 的非法 else
abc E0127 应输入一个语句



§ . 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

Microsoft Visual Studio 调试控制台

```
12 34
max=34
max=34
max=34
max=34
```

2、输入34 12，给出运行截图

Microsoft Visual Studio 调试控制台

```
34 12
max=34
max=34
max=34
max=34
```

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_3>2>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)

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

int main()
{
    int a=1, b=2;

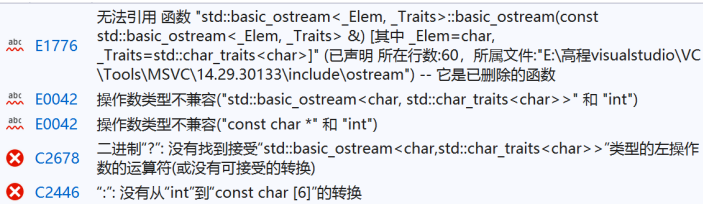
    a==1 ? "Hello" : 123; //编译报错

    a>b ? cout << a : printf("%d", b); //编译报错

    a==1 ? 'A' : 123; //编译正确

    return 0;
}
```

1、给出编译报错的截图



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）

冒号前后表达式的类型必须相容，否则无法确定表达式的值类型



§. 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

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： 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

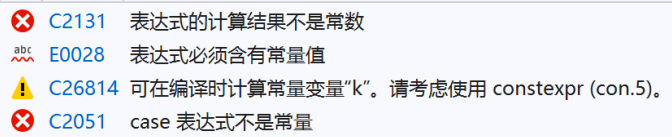
C. 观察下列程序的运行结果，回答问题并将程序的运行结果截图贴上(如果有错则贴错误信息截图)

```
#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



2、解释错误原因

Case后的值只能是整形常量



§. 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)

<pre>#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; }</pre>	<p>在8.A的基础上，多了一个case 4+2</p> <p>1、给出编译错误的截图</p> <div></div> <p>2、解释错误原因</p> <p>case后的表达式的值不能相同，4+2=6, 6重复使用</p>
---	---



§. 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;
    }

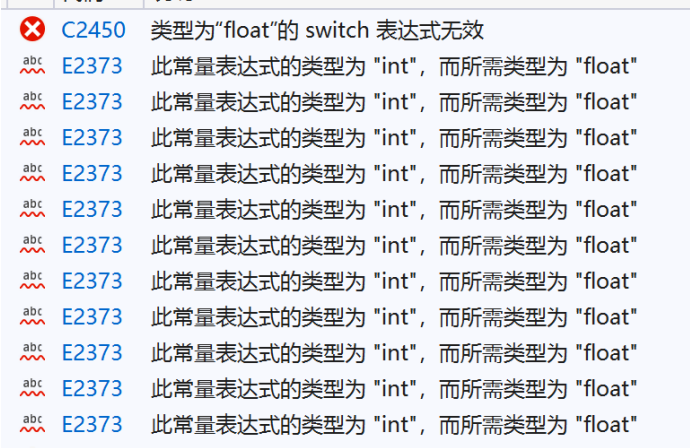
    return 0;
}
```

在8.A的基础上，将score从int改为float

1、给出编译错误的截图

2、解释错误原因

Switch表达式只能是整型






§. 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)

<pre>#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; }</pre>	<p>在8.A的基础上，删除case 8后面的break</p> <p>1、给出与8. A运行结果不一致的测试数据即截图</p>  <p>2、解释break的作用 使程序跳出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

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，即符合预期的期望

```
#include <iostream>
using namespace std;
int main()
{
    int score;
    cout << "请输入成绩[0-100]" << endl;
    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：[85-100] - 优
[70-85) - 良
[60-70) - 及格
[0-60) - 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

If能用，switch不能用。小数形式为实型，而switch括号内只能是整型

4、总结switch语句使用时的注意事项

Switch括号内只能是整型表达式，case后只能为整型常量，值不能重复

5、switch-case语句能完全取代if-else吗？

不能

```
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 < 60) {
10         cout << "不及格" << endl;
11     }
12     else if (score >= 60 && score < 70) {
13         cout << "及格" << endl;
14     }
15     else if (score >= 70 && score < 85) {
16         cout << "良" << endl;
17     }
18     else if (score >= 85 && score <= 100) {
19         cout << "优" << endl;
20     }
21     else {
22         cout << "输入错误" << endl;
23     }
24
25     return 0;
26 }
```



```
switch (score / 5) {
case 20:
case 19:
case 18:
case 17:
    cout << "优" << endl;
    break;
case 16:
case 15:
case 14:
    cout << "良" << endl;
    break;
case 13:
case 12:
    cout << "及格" << endl;
    break;
case 11:
case 10:
case 9:
case 8:
case 7:
case 6:
```



§. 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