



9.3 控制结构

9.4 常用算法

同济大学

9.3 控制结构



➤ 顺序结构

➤ 选择结构

➤ 循环结构



本单元教学目的

- ▶了解三种基本控制**结构**的作用
- ▶**熟练**运用**if**语句的多种分支形式描述程序
- ▶**熟练**运用三种**循环语句**描述各种重复操作
- ▶**领会**复合**语句**在控制**结构**中的正确运用



一、从引例了解三种结构

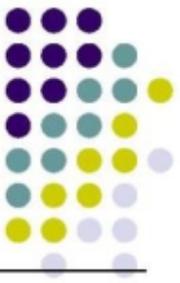
【引例】利用公式

$$s = \sqrt{p(p-a)(p-b)(p-c)}$$

编程求以a、b、c做三边的三角形的面积。其中，p为三角形三边和的一半，即： $p = (a+b+c)/2$

分析：

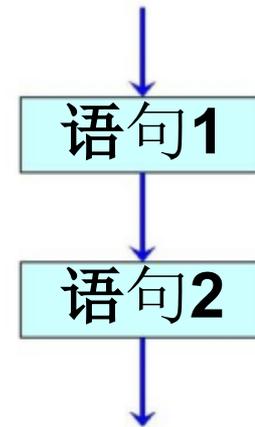
求解问题的数学模型能够通过一个数学公式来描述时程序是最易实现的，按以往的知识可将问题的核心部分表达如下：



引例分析实现

程序段1:

```
cin>>a>>b>>c;  
p=(a+b+c)/2;  
s=sqrt(p*(p-a)*(p-b)*(p-c));  
cout<<"三角形面积为:"<<s<<endl;
```



- 按从上到下的顺序逐语句执行即为**顺序结构**;
- 语句是有顺序的，**顺序结构不是语句的简单罗列**。
(观察以下初学者常犯错误)

```
p=(a+b+c)/2;  
cin>>a>>b>>c;  
s=sqrt(p*(p-a)*(p-b)*(p-c));
```



引例分析实现

运行包含程序段**1**的程序并思考：

? 分别输入两组测试数据**3、4、5**和**3、4、9**，
结果如何？

? 问题产生的原因？

未考虑公式的适用条件

? 如何解决？

有条件地按公式求解



引例分析实现

用**逻辑表达式**表达
三边构成条件 

```
程序段2:  
  
if( a+b>c&&a+c>b&&b+c>a )  
{  
    p=(a+b+c)/2;  
    s=sqrt(p*(p-a)*(p-b)*(p-c));  
    cout<<"三角形面积为:"<<s<<endl;  
}
```

单分支选择结构 

复合语句:
一对花括号将多条语句组合成一个整体, 以满足语法上对此位置只允许有一条语句的要求。 

```
if(表达式)  
    语句
```

值非**0**执行语句, 否则跳过



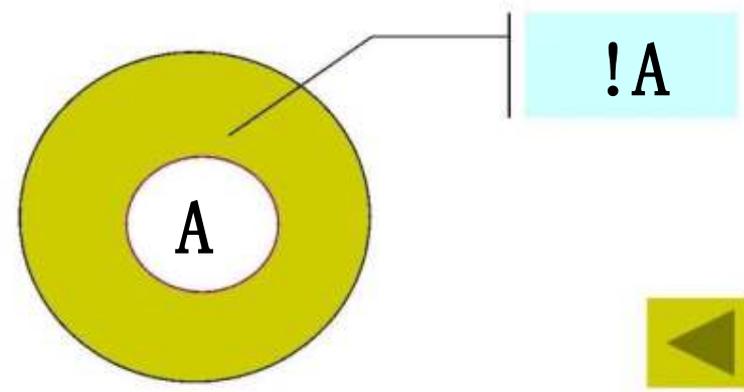
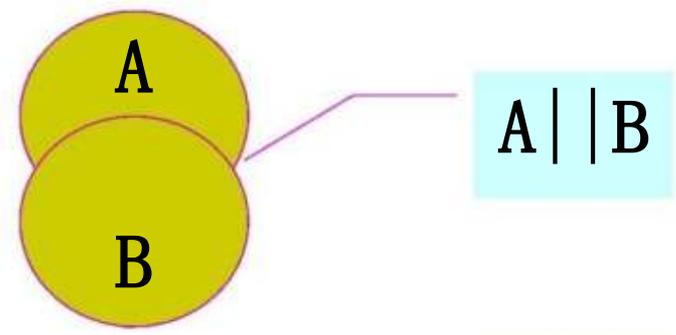
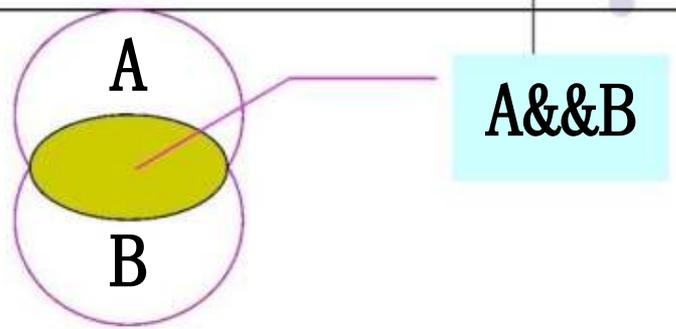
逻辑运算符

种类:

- 逻辑与 (**&&**)
- 逻辑或 (**||**)
- 逻辑非 (**!**)

A	B	A&&B	A B	!A
0	0	0	0	1
0	1	0	1	1
1	0	0	1	0
1	1	1	1	0

0----false 1---true





复合语句

形式：

```
{  
    [变量定义]  
    语句组  
}
```

作用：

当程序中某个位置在**语法**上只允许一条语句，
而在**语义**上要**执行**多条语句才能完成某个操作
时，需要使用复合语句。



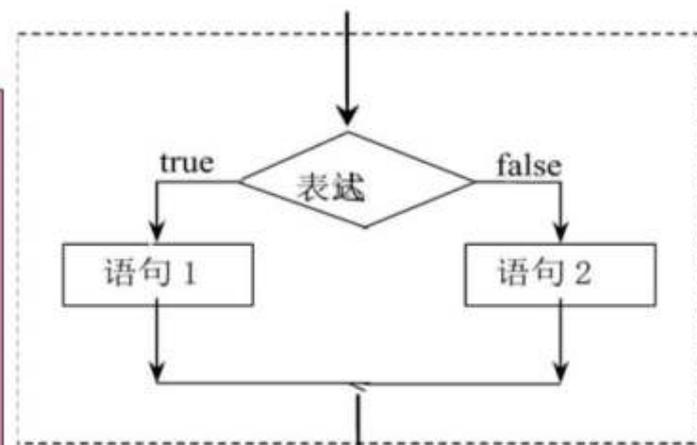


引例分析实现

如何**进**一步改**进**程序段**2**，在不**满**足三角形**三边**构成条件**时**给予相应的提示？

程序段**3**:

```
if (a+b>c&&a+c>b&&b+c>a)
{p=(a+b+c)/2;
 s=sqrt(p*(p-a)*(p-b)*(p-c));
 cout<<"三角形面积为:"<<s<<endl;
}
else
  cout<<"不是三角形的三边"<<endl;
```



双分支选择结构



```
if(表达式)
  语句1
else
  语句2
```

非**0**执行语句**1**，否则执行语句**2**



引例分析实现

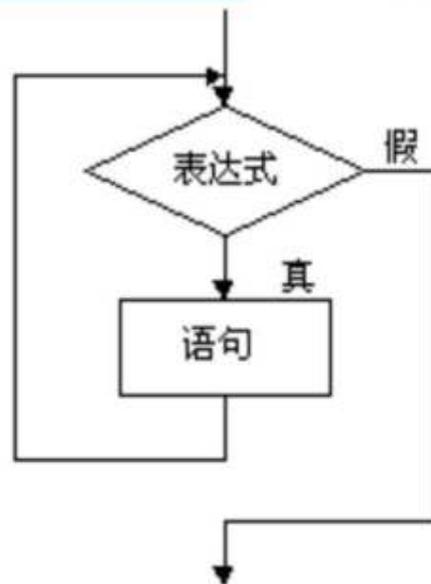
不构成三边

? 进一步思考:

如何在**输入错误数据时**允许重新输入, 以便能得到正确的**结果**?

程序段4:

```
while (! (a+b>c && a+c>b && b+c>a))  
{  
    cout << "数据错误, 重新输入:" << endl;  
    cin >> a >> b >> c;  
}  
p = (a+b+c) / 2;  
s = sqrt(p * (p-a) * (p-b) * (p-c));  
cout << "三角形面积为:" << s << endl;
```



循环结构

while(表达式)
语句

非0反复执行循环体语句, 否则退出循环体



二、选择结构

特点:

根据条件的**值**判断程序的流向。

种类:

➤ if语句

- ✓ 单分支if语句
- ✓ 双分支if-else语句
- ✓ 多分支if-else-if语句

➤ switch语句: 实现多分支 (本课程略去)



典型案例

- 1、比较两数大小1
- 2、比较两数大小2
- 3、成绩转换
- 4、分组体检



if语句单分支形式示例

【例1】对任意输入的两个整数 x ， y 进行处理，要求将大数放在 x 中，小数存于 y 中。

要点：

当 $x < y$ 时需交换两数

```
t=x; x=y; y=t;
```

? 如何交换两个变量

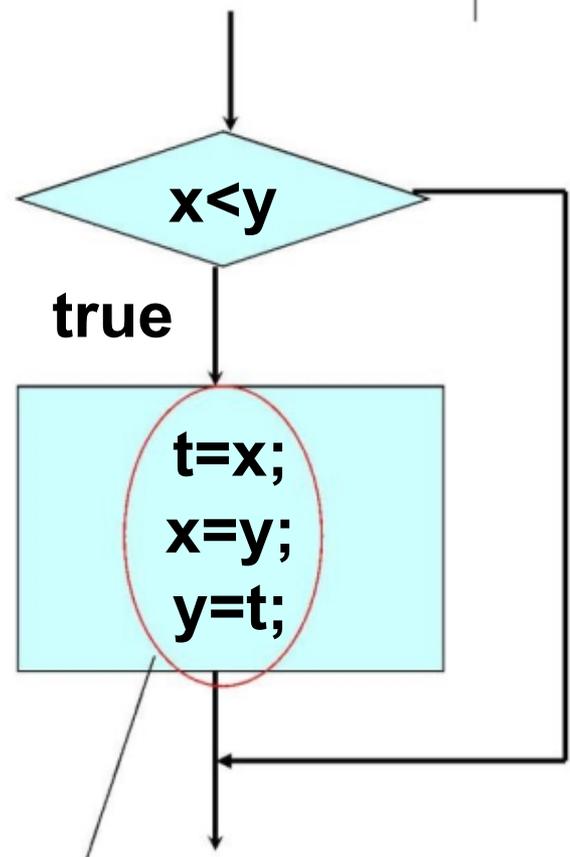
? 如何把交换变量的语句置于某条件控制之下

```
if(x<y)
{ t=x; x=y; y=t; }
```

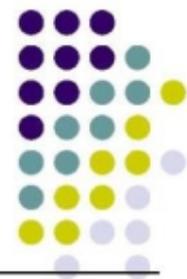


程序实现

```
#include "iostream.h"
void main()
{
    int x, y, t;
    cout<<"入 x y"<<endl;
    cin>>x>>y;
    if (x<y)
    { t=x; x=y; y=t;}
    cout<<x<<">"<<y<<endl;
}
```



三条语句作为一个执行整体



if语句双分支形式示例

【例2】 将任意输入的两个整数 x ， y 中较大的数放在 \mathbf{max} 变量中，较小的数存于 \mathbf{min} 变量中。

要点：

根据 $\mathbf{x < y}$ 成立与否，分别对变量 \mathbf{max} 和 \mathbf{min} 赋不同值。

关键：

多条语句置于分支控制之下的表达



程序实现



```
#include "iostream.h"
void main()
{
    int x, y, max, min;
    cout<<"入 x y"<<endl;
    cin>>x>>y;
    if (x<y)
    { min=x; max=y;}
    else
    { min=y; max=x;}
    cout<<"min="<<min<<endl;
    cout<<"max="<<max<<endl;
}
```

观察并思考：

? 去掉**if**表达式后面的一对花括号会怎样？

? 去掉**else**表达式后面的一对花括号会怎样？

分别对照以下两组数据的结果

第一组： **10 20**

第二组： **20 10**



? 原因



条件运算符

前面的**if-else**语句还可通过**C**语言中的条件运算实现:

```
x < y ? min=x, max=y : min=y, max=x;
```

条件运算符是唯一的三目运算符，形式如下:

```
e1 ? e2 : e3
```

作用:

当**e1**成立时，将**e2**的值作为整个表达式的值，不再计算**e3**的值；否则取**e3**的值作为整个表达式的结果，而不计算**e2**的值。



if语句多分支形式

```
if (表达式1)
    语句1
else if (表达式2)
    语句2
    ⋮
else if (表达式n)
    语句n
else
    语句n+1
```

说明:

任意一个表达式非**0**，则执行其后紧跟的一条语句而跳过所有其它语句；若所有表达式皆为**0**，则执行语句**n+1**。



if语句多分支形式示例

【例3】已知百分制成绩 $mark$ ，要求将其转换为对应的五级制成绩，评定条件：

等级=	{	A	($mark \geq 90$)
		B	($80 \leq mark < 90$)
		C	($70 \leq mark < 80$)
		D	($60 \leq mark < 70$)
		E	($mark < 60$)



要点分析

要点:

- ? 如何正确表达字符**变量**与字符常量
- ? 如何保证**逻辑**关系正确

```
if (mark >= 60)
    grade='D';
else if ( mark >= 70)
    grade='C';
else if (mark >= 80)
    grade='B';
else if (mark >=90)
    grade='A';
else
    grade='E';
```

定义代表等级的**变量**grade,
其**值**根据mark范围分别取
常量'A'~'E'

错误原因?



程序实现

```
#include "iostream.h"
void main()
{
    int mark;
    char grade;
    cout<<"input mark(0~100):"<<endl;
    cin>>mark;
    if(mark >= 90)
        grade='A';
    else if(mark>= 80)
        grade='B';
    else if(mark>=70)
        grade='C';
    else if(mark>=60)
        grade='D';
    else
        grade='E';
    cout<<"mark="<<mark<<" , grade="<<grade<<endl;
}
```

? 思考：若将五级表示为“优、良、中、及格、不及格”，还能使用**grade**变量吗？程序如何修改？



if语句的嵌套形式

【例4】某学校对教工进行分组体检，分组如下：

40岁以下男教工----A组

40岁以上男教工----B组

40岁以下女教工----C组

40岁以上女教工----D组

要点：

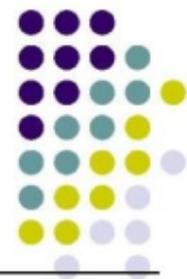
多分支语句实现

？ 按性别和年龄的条件组合，如何表达？

？ 同一性别条件下，按不同年龄分不同组，如何表达？

```
if (sex=='m')
    if (age<40)
        group='A';
    else
        group='B';
```

if后面的语句本身
又是一个if语句



if语句的嵌套形式

在**if**语句中，如果**if**或**else**后面的语句本身又是一个**if**语句，则称为**if**语句的嵌套。如：

```
if(表达式1)
    if(表达式11)
        语句11
    else
        语句12
else
    语句2
```

- ✦锯齿书写格式反映语句间的层次，以增强可读性
- ✦**else**始终与它上面最近的**if**语句配对，但要求这个**if**语句没有其它的**else**与之匹配。



程序实现

```
#include "iostream.h"
void main()
{ char sex, group;
  int age;
  cout<<"input sex(' f' or ' m' ) and age:"<<endl;
  cin>>sex>>age;
  if(sex==' m' )
    if(age<40)
      group=' A' ;
    else
      group=' B' ;
  else
    if(age<40)
      group=' C' ;
    else
      group=' D' ;
  cout<<"group="<<group<<endl;
}
```

if和else本身的语句
又都是一个分支语句

? 将此嵌套形式如何
转换成多分支形式



三 循环结构

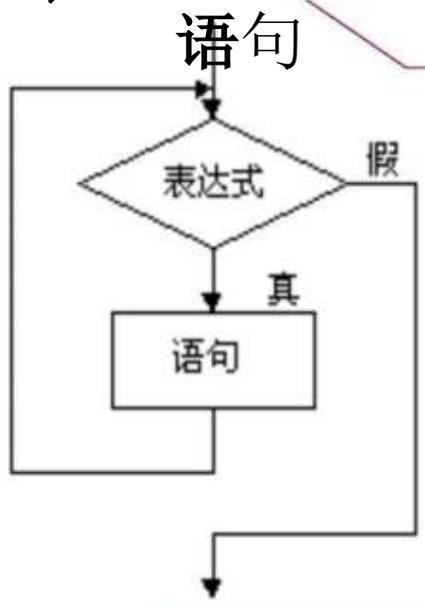
循环条件

当型循环：

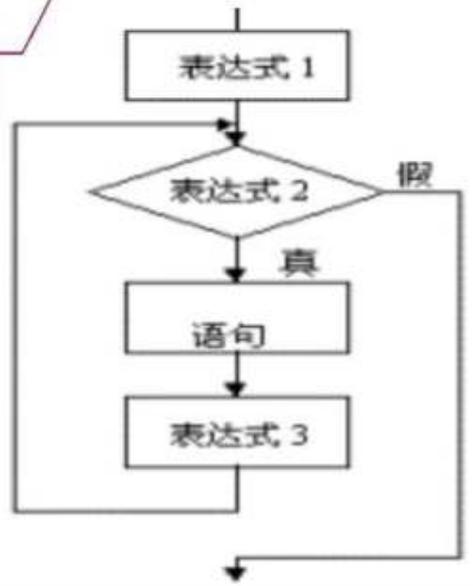
先判断后执行的循环语句（循环体可能一次也不执行）

while (表达式)

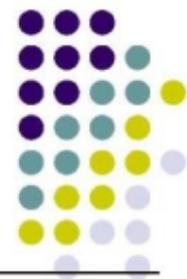
for (表达式1; 表达式2; 表达式3)
语句



循环体
(一条语句)



循环条件非0时执行循环体，否则退出循环



9.4 常用算法和案例

- 1、累加和
- 2、猜数游 戏
- 3、ASCII码对照表
- 4、求极大 值
- 5、最大公约数
- 6、素数
- 7、穷举
法



循环结构示例

【例5】求1~99的奇数和，即 $s=1+3+5+ \dots +99$

要点：

? 如何表达重复求和操作

循环结构

? 什么条件下重复求和

循环条件

? 重复什么**动作**

循环体

? 如何**终止**循环

循环体中包含使循环条件发生改变的语句

? 求和前必要的**准备**

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/408121004006006107>