

第4章 选择结构程序设计

4.1 选择结构和条件判断

4.2 用**if**语句实现选择结构

4.3 关系运算符和关系表达式

4.4 逻辑运算符和逻辑表达式

4.5 条件运算符和条件表达式

4.6 选择结构的嵌套

4.7 用**switch**语句实现多分支选择结构

4.8 选择结构程序综合举例

4.6 选择结构的嵌套

➤ 在**if**语句中又包含一个或多个**if**语句称为**if**语句的嵌套

➤ 一般形式:

if()

if() 语句1
else 语句2

else

if() 语句3
else 语句4

else总是与它上面最近的未配对的**if**配对

内嵌**if**

4.6 选择结构的嵌套

- 在**if**语句中又包含一个或多个**if**语句称为**if**语句的嵌套

if ()

{

if () 语句1

}

else 语句2

内嵌**if**

{ }限定了内嵌**if**范围

4.6 选择结构的嵌套

例**4.5**有一函数：

$$y = \begin{cases} -1 & (x < 0) \\ 0 & (x = 0) \\ 1 & (x > 0) \end{cases}$$

编一程序，输入一个**x**值，要求输出相应的**y**值。

4.6 选择结构的嵌套

➤ 解题思路:

- ◆ 用**if**语句检查**x**的值，根据**x**的值决定赋予**y**的值
- ◆ 由于**y**的可能值不是两个而是三个，因此不可能只用一个简单的(无内嵌**if**)的**if**语句来实现

4.6

➤ 解题思路

```
scanf("%d",&x);  
if(x<0)    y = -1;  
if(x==0)   y = 0;  
if(x>0)    y = 1;  
printf("x=%d,y=%d\n",x,y);
```

(1) 先后用5个独立的if语句处理:

输入x

若 $x < 0$, 则 $y = -1$

若 $x = 0$, 则 $y = 0$

若 $x > 0$, 则 $y = 1$

输出x和y

```
-5  
x=-5,y=-1
```

4.6

➤ 解题思路

(2) 用一

输入 x

若 $x < 0$, 则 $y = -1$

否则

若 $x = 0$, 则 $y = 0$

否则 $y = 1$

输出 x 和 y

```
scanf("%d",&x);  
if(x<0) y=-1;  
else  
    if(x==0) y=0;  
    else y=1;  
printf("x=%d,y=%d\n",x,y);
```

```
-5  
x=-5,y=-1
```

```
5  
x=5,y=1
```

4.6

➤ 解题思路

(2) 用一

输入 x

若 $x < 0$, 则 $y = -1$

```
scanf("%d",&x);
```

```
if(x<0) y=-1;
```

```
else
```

提倡内嵌if放在else中

```
if(x==0) y=0;
```

```
else y=1;
```

```
printf("x=%d,y=%d\n",x,y);
```

```
scanf("%d",&x);
```

```
if (x>=0)
```

```
    if (x>0) y=1;
```

```
    else y=0;
```

```
else y=-1;
```

```
printf("x=%d,y=%d\n",x,y);
```


4.7 用switch语句实现多分支选择结构

➤ **switch**语句用来实现多分支选择结构

◆ 学生成绩分类

85分以上为'**A**'等

70~84分为'**B**'等

60~69分为'**C**'等

.....

◆ 人口统计分类

按年龄分为老、中、青、少、儿童

4.7 用switch语句实现多分支选择结构

例**4.6** 要求按照考试成绩的等级输出百分制分数段，**A**等为**85**分以上，**B**等为**70~84**分，**C**等为**60~69**分，**D**等为**60**分以下。成绩的等级由键盘输入。

➤ 解题思路：

- ◆ 这是一个多分支选择问题
- ◆ 根据百分制分数将学生成绩分为**4**个等级
- ◆ 如果用**if**语句，至少要用**3**层嵌套的**if**，进行**3**次检查判断
- ◆ 用**switch**语句进行一次检查即可得到结果

```
#include <stdio.h>
```

```
int main()
```

```
{ char grade;
```

```
scanf("%c",&grade);
```

```
printf("Your score:");
```

```
switch(grade)  值为A
```

```
{ case 'A': printf("85~100\n");break;
```

```
case 'B': printf("70~84\n");break;
```

```
case 'C': printf("60~69\n");break;
```

```
case 'D': printf("<60\n");break;
```

```
default: printf("enter data error!\n");
```

```
}
```

```
return 0;
```

```
}
```

```
A  
Your score:85~100
```

```
#include <stdio.h>
int main()
{ char grade;
  scanf("%c",&grade);
  printf("Your score:");
  switch(grade)
  { case 'A': printf("85~100\n");break;
    case 'B': printf("70~84\n");break;
    case 'C': printf("60~69\n");break;
    case 'D': printf("<60\n");break;
    default: printf("enter data error!\n");
  }
  return 0;
}
```

不能少

```
#include <stdio.h>
```

```
int main()
```

```
{ char grade;
```

```
scanf("%c",&grade);
```

```
printf("Your score:");
```

```
switch(grade)  值为C
```

```
{ case 'A': printf("85~100\n");break;
```

```
case 'B': printf("70~84\n");break;
```

```
case 'C': printf("60~69\n");break;
```

```
case 'D': printf("<60\n");break;
```

```
default: printf("enter data error!\n");
```

```
}
```

```
return 0;
```

```
}
```

```
C  
Your score:60~69
```

```

#include <stdio.h>
int main()
{ char grade;
  scanf("%c",&grade);
  printf("Your score:");
  switch(grade)  值为F
  { case 'A': printf("85~100\n");break;
    case 'B': printf("70~84\n");break;
    case 'C': printf("60~69\n");break;
    case 'D': printf("<60\n");break;
    default: printf("enter data error!\n");
  }
  return 0;
}

```



A terminal window showing the output of the program. The first line shows 'F' entered as input. The second line shows the output 'Your score:enter data error!'.

```

F
Your score:enter data error!

```

```
#include <stdio.h>
```

```
int main()
```

```
{ char grade;
```

```
scanf("%c",&grade);
```

```
printf("Your score:");
```

此行位置有问题，
应如何修改？

```
switch(grade)
```

```
{ case 'A': printf("85~100\n");break;
```

```
case 'B': printf("70~84\n");break;
```

```
case 'C': printf("60~69\n");break;
```

```
case 'D': printf("<60\n");break;
```

```
default: printf("enter data error!\n");
```

```
}
```

```
return 0;
```

```
F  
Your score:enter data error!
```

```
}
```

➤ **switch**语句的作用是根据表达式的值，使流程跳转到不同的语句

➤ **switch**语句的一般形式：

switch (表达式) 整数类型(包括字符型)

```
{ case 常量1 : 语句1  
  case 常量2 : 语句2  
    |      |  
  case 常量n : 语句n  
  default   : 语句n+1  
}
```


➤ **switch**语句的作用是根据表达式的值，
使流程跳转到不同的语句

➤ **switch**语句的一般形式：

switch (表达式)

```
{ case 常量1 : 语句1  
  case 常量2 : 语句2  
    |      |  
  case 常量n : 语句n  
  default   : 语句n+1  
}
```

不能相同

```
scanf("%c",&grade);  
printf("Your score:");  
switch(grade)
```

```
{ case 'A': printf("85~100\n");  
  case 'B': printf("70~84\n");  
  case 'C': printf("60~69\n");  
  case 'D': printf("<60\n");  
  default: printf("enter data error!\n");  
}
```

```
A  
Your score:85~100  
70~84  
60~69  
<60  
enter data error!
```

```
scanf("%c",&grade);  
printf("Your score:");  
switch(grade)  
{ case 'A': printf("85~100\n");break;  
  case 'B': printf("70~84\n");break;  
  case 'C': printf("60~69\n");break;  
  case 'D': printf("<60\n");break;  
  default: printf("enter data error!\n");  
}
```

```
scanf("%c",&grade);
printf("Your score:");
switch(grade)
{ case 'A':
  case 'B':
    case 'C': printf("60~69\n");break;
    case 'D': printf("<60\n");break;
    default:  printf("enter data error!\n");
}
```



A
Your score:60~69

4.8选择结构程序综合举例

例**4.7** 写一程序，判断某一年是否闰年。

例**4.8** 求 $ax^2 + bx + c = 0$ 方程的解。