



北京大学
PEKING UNIVERSITY

信息科学技术学院

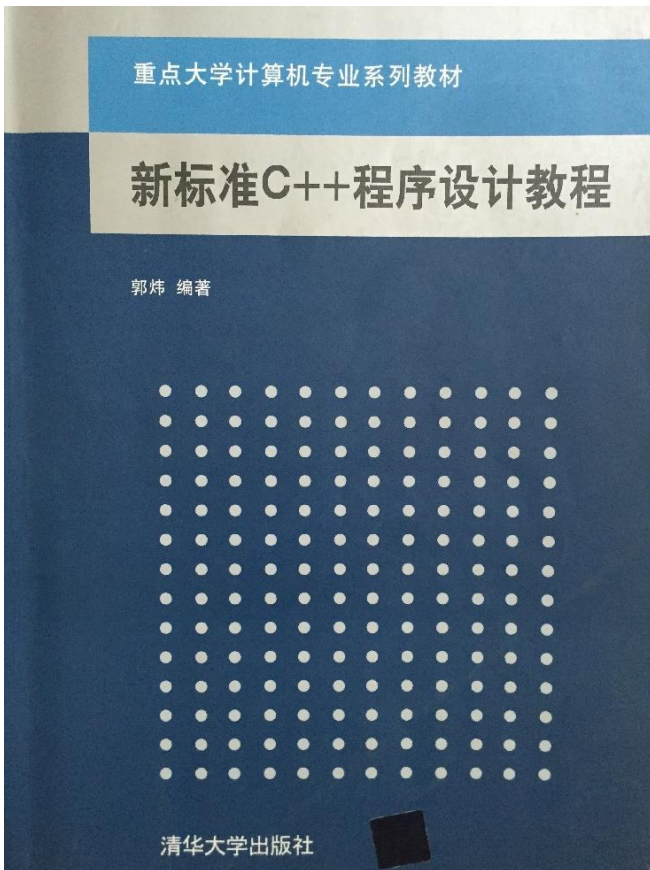
指定教材:

程序设计与算法(一)

李文新 郭炜

主讲教师互动微博:

<http://weibo.com/guoweiofpku>





条件分支结构

switch语句

```
if( n % 5 == 0 ) {  
    .....  
}  
else if( n % 5 == 1 ) {  
    .....  
}  
else if( n % 5 == 2 ) {  
    .....  
}  
else if( n % 5 == 3 ) {  
    .....  
}  
else {  
    .....  
}
```

- 太多的else if 不方便
- $n \% 5$ 多次计算，浪费

switch语句

```
switch(表达式) { //表达式的值 必须是整数类型(int,char .....)  
    case 常量表达式1: //常量表达式必须是整数类型的常量(int,char...)  
        语句组1  
        break;  
    case 常量表达式2:  
        语句组2  
        break;  
    .....  
    case 常量表达式n:  
        语句组n  
        break;  
    default:  
        语句组n+1  
}
```

“表达式”的值等于哪个“常量表达式”，就执行相应的语句组。都不相等，则执行default的语句组。也可以没有default分支

“常量表达式” 里面不能包含变量！

switch语句

例题：请写一个程序，接受一个整数作为输入，如果输入1，则输出“Monday”，输入2，则输出“Tuesday”……输入7，则输出“Sunday”，输入其他数，则输出“Illegal”。

```
#include <iostream>
using namespace std;
int main()
{
    int n;
    scanf("%d",& n);
    switch(n) {
        case 1:
            printf("Monday");
            break;
        case 2:
            printf("Tuesday");
            break;
        case 3:
            printf("Wednesday");
            break;
        case 4:
            printf("Thursday");
            break;
```

```
    case 5:
        printf("Friday");
        break;
    case 6:
        printf("Saturday");
        break;
    case 7:
        printf("Sunday");
        break;
    default:
        printf("Illegal");
}
return 0;
}
```

switch语句

switch语句在进入某个case分支后，会一直执行到第一个碰到的“break;”，即使这个“break;”是在后面的case分支里面。如果没有碰到“break;”，则会向下一直执行到switch语句末尾的“}”，包括“default:”部分的语句组也会被执行。


```
#include <iostream>
using namespace std;
int main()    {
    int n;
    scanf("%d",&n);
    switch(n%6) {
        case 0:
            printf( "case 0" );
            break;
        case 1:
            printf( "case 1" );
        case 2:
        case 3:
            printf( "case 2 or 3" );
            break;
        case 4:
            printf( "case 4" );
            break;
    }
    return 0;
}
```

1✓
case 1
case 2 or 3

2✓
case 2 or 3

3✓
case 2 or 3