



复合算术运算赋值

简写	含义
$a += b$	$a = a + b$
$a -= b$	$a = a - b$
$a *= b$	$a = a * b$
$a /= b$	$a = a / b$
$a \% = b$	$a = a \% b$

表1. 复合算术运算赋值



分卡片

【问题描述】

周末，小智和同学玩分卡片游戏。一共 4 人，小智是裁判，其余 3 人围成一个圈分别编号为 1,2,3，并且每人手里都有若干张卡片（键盘输入）。

现在从 1 号开始将自己的卡片均分成 3 份（如有多余的，交给小智），自己留 1 份，其余两份分给相邻的人，接着 2、3 号同样这么做。

问一轮后，每人（包括小智）手里分别有多少张卡片。

【输入样例】

5 6 8

【输出样例】

6 5 3 5

【算法分析】

①将卡片均分成三份，先把多余部分给到小智。

②然后将均分的三份分别给到左右相邻的人。

每个人（除小智外）依次执行一次①②操作，输出 4 人手里的卡片数。

【参考代码】

```
1. #include<iostream>
2. using namespace std;
3. int main()
4. {
5.     //d 为小智
6.     int a, b, c, d = 0;
7.     cin >> a >> b >> c;
8.     //1 号分卡片
9.     d += (a % 3);
10.    a /= 3;
11.    b += a;
12.    c += a;
13.    //2 号分卡片
```



```
14.     d += (b % 3);
15.     b /= 3;
16.     c += b;
17.     a += b;
18.     //3 号分卡片
19.     d += (c % 3);
20.     c /= 3;
21.     a += c;
22.     b += c;
23.     cout << a << " " << b << " " << c << " " << d << endl;
24.     return 0;
25. }
```



分钱游戏

【问题描述】

甲、乙、丙三人共有 24 元钱。

先由甲分钱给乙、丙两人，所分给的数与各人已有数相同；接着由乙分给甲、丙，分法同前；再由丙分钱给甲、乙，分法亦同前。

经上述三次分钱之后，每个人的钱数恰好一样多。求原先各人的钱数分别是多少？

【运行结果】

13 7 4

【算法分析】

设甲、乙、丙三人的钱数分别为 a , b , c 。

用倒推算法，从最后结果入手，按反向顺序，分步骤推算出每次各人当时的钱数。

【参考代码】

```
1. #include<iostream>
2. using namespace std;
3. int main()
4. {
5.     int a, b, c;
6.     //对应于步骤 1
7.     a = 8;
8.     b = 8;
9.     c = 8;
10.    //对应于步骤 2
11.    a = a / 2;
12.    b = b / 2;
13.    c = a + b + c;
14.    //对应于步骤 3
15.    a = a / 2;
16.    c = c / 2;
17.    b = a + b + c;
18.    //对应于步骤 4
19.    b = b / 2;
20.    c = c / 2;
```



```
21.     a = a + b + c;  
22.     //输出结果  
23.     cout << a << " " << b << " " << c;  
24.     return 0;  
25. }
```



分糖果

【问题描述】

某幼儿园里，有 5 个小朋友编号为 1, 2, 3, 4, 5，他们按自己的编号顺序围坐在一张圆桌旁。他们身上都有若干个糖果（键盘输入），现在他们做一个分糖果游戏。从 1 号小朋友开始，将自己的糖果均分三份（如果有多余的糖果，则立即吃掉），自己留一份，其余两份分给他的相邻的两个小朋友。接着 2 号、3 号、4 号、5 号小朋友同样这么做。问一轮后，每个小朋友手上分别有多少糖果。

【输入样例】

```
8 9 10 11 12
```

【输出样例】

```
11 7 9 11 6
```

【算法分析】

用 a,b,c,d,e 分别存储 5 个小朋友的糖果数。

【参考程序】

```
1. #include<iostream>
2. using namespace std;
3. int main()
4. {
5.     int a, b, c, d, e;
6.     cin >> a >> b >> c >> d >> e;
7.     a/=3; b+=a; e+=a;    //1 号小朋友分糖
8.     b/=3; c+=b; a+=b;    //2 号小朋友分糖
9.     c/=3; d+=c; b+=c;    //3 号小朋友分糖
10.    d/=3; e+=d; c+=d;    //4 号小朋友分糖
11.    e/=3; a+=e; d+=e;    //5 号小朋友分糖
12.    cout << a << " " << b << " " << c << " " << d << " " << e << endl;
13.    return 0;
14. }
```