

250708大促销（题解）

题目链接：

<https://www.matiji.net/exam/brushquestion/34/4009/C448715ED43BEA9D2D47CED523050945?from=1>

小码哥门口的西瓜摊开始搞大促销了。摊位一旁的牌子上写着：“买 X 个西瓜赠送 Y 个免费的西瓜”。也就是说你每买 X 个西瓜，摊位老板都会给你 Y 个免费的西瓜。为了简化问题，我们现在假设每个西瓜的重量相同，且购买单个西瓜的都需要花费 a 的代价。现在小码哥想购买一些西瓜，请你帮他计算一下最少需要花费多少的代价。

输入格式：

输入一行四个整数 X, Y, a, N ($1 \leq X, Y, a \leq 1000, 1 \leq N \leq 10000$)，表示西瓜购买赠送的比例，单个西瓜的价格为 a ，以及小码哥希望购买 N 个西瓜。

输出格式：

输出一个整数，表示最小花费的代价。

样例1

输入：2 1 2 10

输出：14

解法一：模拟

```
#include<bits/stdc++.h>
using namespace std;

int main() {
    int x, y, a, n;
    cin >> x >> y >> a >> n;

    int sum = 0;
    // 每次买x个西瓜时都需要花费x*a的代价
    for(int i = 1; i <= n; ++i) {
        sum += a; // 每次购买西瓜都加上购买的代价
        if(i % x == 0) { // 如果已经买了x个西瓜
            n -= y; // 赠送y个西瓜，相当于减少了y个购买的需求
        }
    }
    cout << sum; // 输出总花费
    return 0;
}
```

解法二：数学推导

```
#include <bits/stdc++.h>
using namespace std;

int main() {
    int X, Y, a, N;
    cin >> X >> Y >> a >> N;

    int groupTotal = X + Y; // 每轮购买获得的西瓜数（包括赠品）
    int totalGroup = N / groupTotal; // 完整轮数
    int remaining = N % groupTotal; // 剩余需要购买的西瓜数

    // 总花费是完整轮数 * 每轮买的西瓜数 * 单价
    int cost = totalGroup * X * a;

    // 如果剩余西瓜数大于0，则计算剩余需要购买的西瓜
    if (remaining > 0) {
        cost += min(remaining, X) * a;
    }

    cout << cost << endl;
    return 0;
}
```

题目拓展：

加一个条件“赠送的西瓜+新买的西瓜每满 x 个就再送 y 个免费的西瓜”，其他要求不变

例如：输入：3 1 5 10

输出：35

解释：买三个瓜送一个，此时手里有四个（此时实际买了三个）

再买两个瓜+上面送的一个满三个，再送一个瓜，此时手里有七个（此时实际买了五个）

再买两个瓜+上面送的一个满三个，再送一个瓜，此时手里有十个（此时实际买了七个）

题目本质：

除了第一轮买 x 送 y

从第二轮开始实际是买 $x-y$ 送 y

#参考代码

```
#include <bits/stdc++.h>
using namespace std;

int main() {
    int X, Y, a, N;
    cin >> X >> Y >> a >> N;
```

```

int now = 0; // 当前西瓜总数
int cost = 0; // 总花费

if (N <= X) {
    cost = N * a;
} else if (N <= X + Y) {
    cost = X * a;
} else {
    // 第一轮
    cost = X * a;
    now = X + Y;

    while (now < N) {
        int remaining = N - now;
        int nextBuy = min(X - Y, remaining); // 只买需要的
        cost += nextBuy * a;
        now += nextBuy;

        // 赠送逻辑
        int total = now % X;
        if (total == 0) {
            now += Y;
        }
    }
}

cout << cost << endl;
return 0;
}

```