

P1208 混合牛奶

题目描述

由于乳制品产业利润很低，所以降低原材料（牛奶）价格就变得十分重要。帮助Marry乳业找到最优的牛奶采购方案。

Marry乳业从一些奶农手中采购牛奶，并且每一位奶农为乳制品加工企业提供的价格是不同的。此外，就像每头奶牛每天只能挤出固定数量的奶，每位奶农每天能提供的牛奶数量是一定的。每天Marry乳业可以从奶农手中采购到小于或者等于奶农最大产量的整数数量的牛奶。

给出Marry乳业每天对牛奶的需求量，还有每位奶农提供的牛奶单价和产量。计算采购足够数量的牛奶所需的最小花费。

输入格式

第 1 行共二个数值: N , ($0 \leq N \leq 2,000,000$)是需要牛奶的总数; M , ($0 \leq M \leq 5,000$)是提供牛奶的农民个数。

第 2 到 $M+1$ 行: 每行二个整数: P_i 和 A_i 。

P_i ($0 \leq P_i \leq 1,000$) 是农民 i 的牛奶的单价。

A_i ($0 \leq A_i \leq 2,000,000$) 是农民 i 一天能卖给Marry的牛奶制造公司的牛奶数量。

输出格式

单独的一行包含单独的一个整数，表示Marry的牛奶制造公司拿到所需的牛奶所要的最小费

输入样例

```
100 5
5 20
9 40
3 10
8 80
6 30
```

输出样例

```
630
```

解析

因为想求最小花费，因此，我们需要按照单价由小到大进行排序，然后购买牛奶。不能整份购买的，可以进行拆分。

编码

```
#include <bits/stdc++.h>
using namespace std;
const int Maxn = 5000 + 1;
int n, m, countNum = 0, maxLength = 0, start, end;
struct Cow {
    int price;
    int num;
};
Cow Cows[Maxn]; //牛奶数组
//按照单价进行排序
bool Compare(Cow a, Cow b) {
    return a.price < b.price;
}
int main() {
    scanf("%d %d", &n, &m);
    for (int i = 1; i <= m; ++i) {
        scanf("%d %d", &Cows[i].price, &Cows[i].num);
        //printf("%d %d\n", Cows[i].price, Cows[i].num);
    }
    sort(Cows + 1, Cows + m + 1, Compare);
    //尝试购买牛奶
    for (int i = 1; i <= m; ++i) {
        Cow cow = Cows[i];
        //整份买
        if (n >= cow.num) {
            countNum += cow.price * cow.num;
            n -= cow.num;
        }
        //部分买
        else if (n > 0) {
            countNum += n * cow.price;
            n = 0;
        }
    }
    cout << countNum;
    return 0;
}
```

逻辑航线培优教育，信息学奥赛培训专家。

扫码添加作者获取更多内容。



