

P1068 [NOIP2009 普及组] 分数线划定

题目描述

世博会志愿者的选拔工作正在 A 市如火如荼的进行。为了选拔最合适的人才，A 市对所有报名的选手进行了笔试，笔试分数达到面试分数线的选手方可进入面试。面试分数线根据计划录取人数的150%划定，即如果计划录取 m 名志愿者，则面试分数线为排名第 $m \times 150\%$ （向下取整）名的选手的分数，而最终进入面试的选手为笔试成绩不低于面试分数线的所有选手。

现在就请你编写程序划定面试分数线，并输出所有进入面试的选手的报名号和笔试成绩。

输入格式

第一行，两个整数 n, m ($5 \leq n \leq 5000, 3 \leq m \leq n$)，中间用一个空格隔开，其中 n 表示报名参加笔试的选手总数， m 表示计划录取的志愿者人数。输入数据保证 $m \times 150\%$ 向下取整后小于等于 n 。

第二行到第 $n+1$ 行，每行包括两个整数，中间用一个空格隔开，分别是选手的报名号 k ($1000 \leq k \leq 9999$) 和该选手的笔试成绩 s ($1 \leq s \leq 100$)。数据保证选手的报名号各不相同。

输出格式

第一行，有2个整数，用一个空格隔开，第一个整数表示面试分数线；第二个整数为进入面试的选手的实际人数。

从第二行开始，每行包含2个整数，中间用一个空格隔开，分别表示进入面试的选手的报名号和笔试成绩，按照笔试成绩从高到低输出，如果成绩相同，则按报名号由小到大的顺序输出。

输入样例

```
6 3
1000 90
3239 88
2390 95
7231 84
1005 95
1001 88
```

输出样例

```
88 5
1005 95
2390 95
1000 90
1001 88
3239 88
```

解析

解题步骤如下：

- 1、计算面试录取人数
- 2、排序所有人笔试成绩
- 3、划定面试分数线
- 4、统计最终面试人数

编码

```
#include<bits/stdc++.h>

using namespace std;
//前面是头文件
struct People {
    int num;        //号码
    int score;      //成绩
};

//自定义排序
int cmp(People p1, People p2) {
    //成绩相同时按照序号进行排序
    if (p1.score == p2.score) {
        //按序号进行排列
        return p1.num < p2.num;
    }
    //否则按照分数进行排序
    else {
        return p1.score > p2.score;
    }
}

int main() {
    int n, m;
    cin >> n >> m;
    //先计算出需要录取的数量
    m = floor(m * 1.5);
    //向下取整
    People people[n];
```

```

    for (int i = 0; i < n; i++) {
        cin >> people[i].num >> people[i].score;
    }
    //进行排序
    sort(people, people + n, cmp);
    //计算出分数线
    //因为索引是从0开始，所以最后一名的成绩是m-1
    int scoreLine = people[m - 1].score;
    //先初步记录一下入选人数
    int sum = m;
    //因为在排名较低的选手中很可能存在分数相同的选手，需要再次进行判断
    for (int i = m; i < n; i++) {
        //存在达到分数线标准的选手则进行记录
        if (people[i].score >= scoreLine) {
            sum += 1;
        }
        //因为分数都是由高到低，依次递减的
        //所以，一旦出现一个低于分数线的选手则说明后面全部不满足
        else if (people[i].score < scoreLine) {
            break;
        }
    }
    //打印最终结果
    cout << scoreLine << " " << sum << endl;
    for (int i = 0; i < sum; i++) {
        cout << people[i].num << " " << people[i].score << endl;
    }
    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。

