

1839: 【05NOIP提高组】谁拿了最多奖学金

题目描述

某校的惯例是在每学期的期末考试之后发放奖学金。发放的奖学金共有五种，获取的条件各不相同：

1) 院士奖学金，每人8000元，期末平均成绩高于80分（ >80 ），并且在本学期内发表1篇或1篇以上论文的学生均可获得；

2) 五四奖学金，每人4000元，期末平均成绩高于85分（ >85 ），并且班级评议成绩高于80分（ >80 ）的学生均可获得；

3) 成绩优秀奖，每人2000元，期末平均成绩高于90分（ >90 ）的学生均可获得；

4) 西部奖学金，每人1000元，期末平均成绩高于85分（ >85 ）的西部省份学生均可获得；

5) 班级贡献奖，每人850元，班级评议成绩高于80分（ >80 ）的学生干部均可获得；

只要符合条件就可以得奖，每项奖学金的获奖人数没有限制，每名学生也可以同时获得多项奖学金。例如姚林的期末平均成绩是87分，班级评议成绩82分，同时他还是一位学生干部，那么他可以同时获得五四奖学金和班级贡献奖，奖金总数是4850元。

现在给出若干学生的相关数据，请计算哪些同学获得的奖金总数最高（假设总有同学能满足获得奖学金的条件）。

输入格式

第一行是一个整数 N （ $1 \leq N \leq 100$ ），表示学生的总数。接下来的 N 行每行是一位学生的数据，从左向右依次是姓名，期末平均成绩，班级评议成绩，是否是学生干部，是否是西部省份学生，以及发表的论文数。姓名是由大小写英文字母组成的长度不超过20的字符串（不含空格）；期末平均成绩和班级评议成绩都是0到100之间的整数（包括0和100）；是否是学生干部和是否是西部省份学生分别用一个字符表示，Y表示是，N表示不是；发表的论文数是0到10的整数（包括0和10）。每两个相邻数据项之间用一个空格分隔。

输出格式

三行，第一行是获得最多奖金的学生的姓名，第二行是这名学生获得的奖金总数。如果有两位或两位以上的学生获得的奖金最多，输出他们之中在输入文件中出现最早的学生的姓名。第三行是这 N 个学生获得的奖学金的总数。

输入样例

```
4
YaoLin 87 82 Y N 0
ChenRuiyi 88 78 N Y 1
LiXin 92 88 N N 0
ZhangQin 83 87 Y N 1
```

输出样例

ChenRuiyi
9000
28700

解析

很简单的模拟题，直接按照题目要求的进行计算即可。

编码

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

using namespace std;

//结构体：自定义数据类型
struct node {
    string name; //学生姓名
    int qmf; //期末平均成绩
    int s2; //班级评议成绩
    char p1, p2; //是否是学生干部，是否是西部省份学生
    int lws; //论文数
    //*****以下两个数据并非来自输入*****//
    int sum; //获得的奖学金
    int index; //序号，当奖学金数目一样时，序号小的先输出
};

//学生信息数组
node stu[110];

//学生的数量和最终的奖金数量
int n, ans = 0;

//自定义的排序函数
//a和b比较，当比较条件为真时，a排在前面
bool cmp(node a, node b) {
    //按照奖学金排
    if (a.sum != b.sum) {
        //当a的奖学金大于b的奖学金时，a排在b的前面
        return a.sum > b.sum;
    } else {
        //如果奖学金相同按照序号排
        return a.index < b.index;
    }
}

int main() {
    //读入学生的基本信息
```

```

cin >> n;
for (int i = 1; i <= n; i++) {
    //连续输入学生的各个信息
    cin >> stu[i].name >> stu[i].qmf >> stu[i].s2 >> stu[i].p1 >>
        stu[i].p2 >> stu[i].lws;
    //判断该生是否获得了院士奖学金
    if ((stu[i].qmf > 80) && (stu[i].lws >= 1)) {
        stu[i].sum += 8000;
    }
    if (stu[i].qmf > 85 && stu[i].s2 > 80) {
        stu[i].sum += 4000; //五四奖学金
    }
    if (stu[i].qmf > 90) {
        stu[i].sum += 2000; //成绩优秀奖
    }
    if (stu[i].p2 == 'Y' && stu[i].qmf > 85) {
        stu[i].sum += 1000; //西部奖学金
    }
    if (stu[i].s2 > 80 && stu[i].p1 == 'Y') {
        stu[i].sum += 850; //班级贡献奖
    }
    //记录学生的序号
    stu[i].index = i;
    //ans为N个学生获得的奖学金的总数
    ans += stu[i].sum;
}
//自定义排序，左闭右开
sort(stu + 1, stu + 1 + n, cmp);
//输出最终结果
cout << stu[1].name << endl << stu[1].sum << endl << ans;
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。

