

愉快的代码【7.30】

【7.27】我们特意加入了题解系统，也就是说，姐姐也会重新做一遍给你们的题目（包括选做），然后在第二天的题目前给出姐姐自己的代码和注释作为题解或参考

如果觉得自己的代码略为臃肿，可以参考对比一下姐姐的代码；

如果觉得姐姐的代码不如自己的优秀，也可以尽情地嘲讽姐姐～

【7.30】我们特意加入了团队系统，因为感觉到你们有点像是独立学习的样子，比如说姐姐和你们之间有交流，但是你们之间有没有交流呢姐姐就感受不到啦

所以正好在洛谷上发现了一个团队系统，我们可以在这上面布置作业呀（当然姐姐也会继续以 pdf 形式布置作业，你们也还是要以 pdf 形式交作业哈），然后你们就可以在上面看到其它妹妹们的代码呀（包括 AC 代码和还未 AC 的整个过程的代码和分数呀），觉得她们表现不够自己好的话，就可以在群里尽情地嘲笑她们呀～

然后那上面还有一个比赛功能哇，具体形式和我们平时的机考差不多，暑假差不多结束了我们也会有一次期末模拟机考的哈～

如果你们开心的话，你们也可以联合起来给姐姐布置一次平时的作业呀，或者给姐姐安排一次机考呀，你们都是团队的管理员了哈

7.29问题1：

```
/*
    洛谷P1093:奖学金
    思想：简单的结构体排序，声明一个结构体，记录小学生们的各项数据，然后按照题意排序就好啦
    时间复杂度：O(nlogn)
*/

#include <iostream>
#include <algorithm>
#define ll long long
using namespace std;

struct Node { // 结构体
    int number; // 学号
    int yuwen;
    int shuxue;
    int yinyu;
    int sum;
};

int cmp(Node a, Node b) { // 返回1则a应该排在前面，返回0则b应该排在前面
    if (a.sum != b.sum) return a.sum > b.sum; // 总分大的排前面
```

```

        else if (a.yuwen != b.yuwen) return a.yuwen > b.yuwen; // 总分相同，语文好的在前面
    }
    else return a.number < b.number; // else学号小的在前面
}

int main() {
    int i,j,n;
    cin >> n;
    Node a[n];
    for (i=0; i<n; i++) {
        cin >> a[i].yuwen >> a[i].shuxue >> a[i].yinyu;
        a[i].sum = a[i].yuwen+a[i].shuxue+a[i].yinyu; // 总分
        a[i].number = i+1; // 学号
    }
    sort(a,a+n,cmp); // 结构体排序
    for (i=0; i<5; i++) cout << a[i].number << " " << a[i].sum << endl;
}

```

7.29问题2:

```

/*
    洛谷P1067:多项式输出
    思想：之前说朴实无华也能解，后来发现朴实无华太难解了；比如说我下方注释部分cmp是我之前想到的朴实无华硬比较，虽然能过洛谷，但是还是能随便hack掉（比如说它会错误地把123排在12313的前面）
    还是直接来个巧妙的方式叭，你们做的过程中应该发现直接简单贪心地按照字典序排序是不行滴（比如说123<1231，但是123应该在1231前面），但是我们可以按照合成后的结果的字典序来比较（也就是复杂一点的贪心）：比如说两个字符串a和b，如果是a排在b前面，那么它们合成的字符串就是a+b；如果是b排在a前面，那么它们合成的字符串就是b+a。因为最后要输出最大的整数，因此最后的结果字典序应该是最大。那我们比较a+b和b+a，哪个字典序大就把哪个排在前面啦
    更细致的证明可能需要附带一些玄学的思考，不过贪心题嘛代码简单证明难
    时间复杂度：O(m*n*logn)，m为最长字符串长度
*/

```

```

#include <iostream>
#include <algorithm>
#define ll long long
using namespace std;

/*
int cmp(string a,string b) {
    int i;
    if (a == b) return 1;
    for (i=0; i<a.length() and i<b.length(); i++) {
        if (a[i] > b[i]) return 1;
        else if (a[i] < b[i]) return 0;
    }
    if (i == a.length() and a[0] > b[a.length()]) return 1;
    else if (i == b.length() and a[b.length()] > b[0]) return 1;
}

```

```

        else return 0;
    }
    */

    int cmp(string a,string b) {
        return a+b>b+a; // 合成后字典序大的排前面
    }

    int main() {
        int i,j,n;
        cin >> n;
        string a[n];
        for (i=0; i<n; i++) cin >> a[i];
        sort(a,a+n,cmp);
        for (i=0; i<n; i++) cout << a[i];
    }

```

今天的题目：

知识点：暴力/枚举

“山不在高，有仙则名；码不在快，能过就行”

我们知道有一种解题的方式：把所有的可能性列出来，然后找到满足题意的那一个

这种方法是肯定不会 WA 的（但是可能会 TLE）

不过有一些题，它就得枚举所有情况，然后再挑出合适的

1、<https://www.luogu.com.cn/problem/P1157>

（搜索是一种优雅的暴力

（一些小提示：#include <iomanip>, cout << right << setw(3)

2、<https://www.luogu.com.cn/problem/P3392>

（数据范围足够小， $O(n^3)$ 也能行