

PAT_A1006.Sign In and Sign Out (25)

PAT_A1006.Sign In and Sign Out (25)

- 1.Abstraction
 - 1.1 Method and idea
 - 1.2 Notice
- 2. Problem
- 3. Code
 - 3.1 Edit 0
 - 3.1.1 Algorithm abstraction
 - 3.1.2 Notice

1.Abstraction

1.1 Method and idea

* None

1.2 Notice

* None

2. Problem

At the beginning of every day, the first person who signs in the computer room will unlock the door, and the last one who signs out will lock the door. Given the records of signing in's and out's, you are supposed to find the ones who have unlocked and locked the door on that day.

Input Specification:

Each input file contains one test case. Each case contains the records for one day. The case starts with a positive integer M, which is the total number of records, followed by M lines, each in the format:

```
ID_number Sign_in_time Sign_out_time
```

where times are given in the format HH:MM:SS, and ID number is a string with no more than 15 characters.

Output Specification:

For each test case, output in one line the ID numbers of the persons who have unlocked and locked the door on that day. The two ID numbers must be separated by one space.

Note: It is guaranteed that the records are consistent. That is, the sign in time must be earlier than the sign out time for each person, and there are no two persons sign in or out at the same moment.

Sample Input:

```
3
CS301111 15:30:28 17:00:10
SC3021234 08:00:00 11:25:25
CS301133 21:45:00 21:58:40
```

Sample Output:

```
SC3021234 CS301133
```

3. Code

3.1 Edit 0

3.1.1 Algorithm abstraction

读入，排序，输出，排序，输出

3.1.2 Notice

* None

```
#include <cstdio>
#include <algorithm>
#include <string>
using std::string;
using std::sort;
const int maxsize=20;
struct record{
    char id_num[maxsize];
    int sign_in;
    int sign_out;
};
const int maxn=100010;

record records[maxn];

bool cmp1(record r1,record r2){
    return r1.sign_in<r2.sign_in;
}
bool cmp2(record r1,record r2){
    return r1.sign_out>r2.sign_out;
}
int main(){
    int num;
    scanf("%d",&num);
    int HH,MM,SS;
    for(int i=0;i<num;++i){
```

```
        scanf("%s", records[i].id_num);
        scanf("%d:%d:%d", &HH, &MM, &SS);
        records[i].sign_in=HH*60*60+MM*60+SS;
        scanf("%d:%d:%d", &HH, &MM, &SS);
        records[i].sign_out=HH*60*60+MM*60+SS;
    }
    sort(records, records+num, cmp1);
    printf("%s ", records[0].id_num);
    sort(records, records+num, cmp2);
    printf("%s\n", records[0].id_num);
    return 0;
}
```