

問題B, 審判は君だ!

原案: 川中

解答: 西田, 八森

解説: 八森

問題概要

- プログラミングコンテストのログが与えられます
- ログの内容より、チームの順位表を作って下さい

例

与えられるログの内容

3 3 5

3 1 800 WRONG

1 1 1200 CORRECT

3 1 1400 CORRECT

1 2 2400 CORRECT

1 3 3600 CORRECT

- チーム数、問題数、ログのレコード数
- 時刻800に、チーム3が問題1に誤答
- 時刻1200に、チーム1が問題1に正解

出力するチーム順位表

1 3 7200

3 1 2600

2 0 0

- チーム番号、正解数、ペナルティ

解法

(1) ログより各チームの正解数とペナルティを計算

- 正解数: CORRECTの数をカウント
- ペナルティ: (CORRECTを出した問題の,以下の総和を計算)
CORRECTを出した時の時間+ 1200*その問題の誤答数

(2) 問題文記載のルールに従い、チームを並び替え、出力

- 正解数が多いチームが順位が上
- 正解数が同じなら、ペナルティが少ないチームが順位が上
- 正解数もペナルティも同じなら、チーム番号が小さいチームが順位が上

工夫

- チームの並び替え処理は、ライブラリのソート関数を使うと簡潔に書けます

ライブラリを使えば、自分でソートアルゴリズムを実装する手間が省ける！

• この問題に限らず、プログラミングコンテストではソートは頻出なので、簡潔な書き方を覚えておくと吉！

ソート関数の使用例(C++での実装例)

```
class Team{
public:
    int id;          // チーム番号
    int nCorrect;   // 正解数
    int penalty;    // ペナルティ

    bool operator< (const Team &opp) const { // チームの順位付けルール
        if(nCorrect!=opp.nCorrect) return nCorrect>opp.nCorrect;
        if(penalty!=opp.penalty) return penalty<opp.penalty;
        return id<opp.id;
    }
};
```

```
int nTeam;          // チーム数
Team team[50];     // チームの情報
...                // 与えられるログに基づき、正解数・ペナルティを計算。コードは省略
sort(team, team+nTeam); // ライブラリのソート関数を使ってチームの並べ替え!
```

結果

- 最速正解チーム: wakaba (1506秒)
 - ただし、サーバトラブルのため参考
- 正解チーム数: 93チーム