JAG模擬国内予選2021
A: Tokyo2020

原案:	Darsein
問題文:	tsutaj
データセット:	Darsein
解答:	cos, Darsein, not, riantkb, tsutaj

問題概要

メダルを取った年 Y_i・競技S_i・色M_iの組がN個与えられる。

最も多くの金メダルを取った年と、最も多くのメダル (色問わず) を取った年を出力せよ。
(複数ある場合は最初の年)

制約

- $1 \leq N \leq 200$
- $1896 \leq Y_i \leq 2021$

解法例 (メダル集計パート)

- 年をキーにしたデータ構造にメダルの個数を記録
  - 例えば配列やマップなど
  - 例:

```
int gold_medal[2022], total_medal[2022];
for (int i = 0; i <= 2021; i++) {
    gold_medal[i] = total_medal[i] = 0;
}
for (int i = 0; i < N; i++) {
    if (M[i] == "Gold") gold_medal[Y[i]]++;
    total_medal[Y[i]]++;
}
```

  - S_iは必要ないデータなので無視する

解法例 (最大個数計算パート)

- 年でループして、最大個数とその年を管理する変数を更新
  - 総計メダルの例：
    ```
    int max_medal = 0, max_year = 0;
    for (int i = 0; i <= 2021; i++) {
        if (max_medal < total_medal[i]) {
            max_medal = total_medal[i];
            max_year = i;
        }
    }
    ```
  - 判定に小なりイコール (<=) ではなく小なり (<) を使うことで、同じ個数の時は更新されないため最も早い年を上書きせずに管理できる
  - 金メダルも同様に管理できる

計算量

- メダル集計パート:
  年をキーにしたデータ構造の管理に $O(\max(Y_i))$、データの集計に$O(N)$
- 最大個数計算パート:
  年でループするのに $O(\max(Y_i))$
- トータルで $O(\max(Y_i) + N)$
  - 工夫すれば $O(\max(Y_i) - \min(Y_i) + N)$ に落とせるが、今回は計算量が余裕なので簡単な実装を選んでもOK

注意点

- 数値と文字列が入り混じった入力処理は、言語によっては少し複雑になるのでチェックしておく
- S_iは必要ないデータなので無視する
- 今回必ず最大値が1以上になることが保証されているが、保証されていない場合は初期値に気をつけないと0年を出力してしまうかもしれない
  - 制約をきちんと読み、丁寧に実装すべきか手抜きできるかを判断できるようになるのも大切

ジャッジ解

cos: 58行 (C++)

Darsein: 60行 (C++)

not: 49行 (C++)

riantkb: 25行 (Python)

tsutaj: 32行 (C++)

統計情報

- AC / Trying teams
  - 114/114 (100.00%)
- First Acceptance
  - SPJ (00:04:04)