JOIG 春合宿 1-1
リレー競技

解説：熊田順一

問題概要

異なる 3 人を選んで300mリレーのチームを作る。

3人の100m走のタイム(タイム A)を走る順に A_1, A_2, A_3 (ミリ秒)

バトンパスのタイム(タイム B) を B_1, B_2, B_3 (ミリ秒)とすると

記録は $A_1 + \max(B_1, B_2) + A_2 + \max(B_2, B_3) + A_3$ (ミリ秒)

適切に走る人と順番を決めたとき、記録は最短で何ミリ秒？

入出力例

入力例

番号	1	2	3	4
タイムA	1070	1080	1050	1020
タイムB	90	70	60	100

出力例：3320

例えば部員4,3,2 の順に走らせると、
記録は 1020 + max(100, 60) + 1050 + max(70, 60) + 1080 = 3320 で最短になる。

小課題 1 ($N \le 100$, 25点)

走る人の選び方や走る順番を全部調べる。

三重ループを使って、一番目・二番目・三番目に走る人を決める。
同じ人を二回以上選ばないように注意する。

計算量は $O(N^3)$

小課題1 回答例

```cpp
#include <bits/stdc++.h>
using namespace std;
int main() {
    int N;
    scanf("%d", &N);
    vector<int> A(N), B(N);
    for (int i = 0; i < N; i++) {
        scanf("%d %d", &A[i], &B[i]);
    }
    int ans = 1e9 + 7;
    for (int a = 0; a < N; a++) {
        for (int b = 0; b < N; b++) {
            for (int c = 0; c < N; c++) {
                if (a == b || b == c || c == a) {
                    continue;
                }
                int sum = A[a] + max(B[a], B[b]) + A[b] + max(B[b], B[c]) + A[c];
                ans = min(ans, sum);
            }
        }
    }
    printf("%d¥n", ans);
}
```

小課題2 ($N \leq 3000$, 33点)

走る3人を決めた時の、記録の最小値を考える。

タイム B が短い順に人 1, 2, 3 として、

タイム Aとタイム Bをそれぞれ A_1, A_2, A_3 , B_1, B_2, B_3 とすると、

- 二番目に走る人が人 1 の時、
  - 記録は $A_1 + A_2 + A_3 + \max(B_2, B_1) + \max(B_1, B_3) = A_1 + A_2 + A_3 + B_2 + B_3$
- 二番目に走る人が人 2 の時、
  - 記録は $A_1 + A_2 + A_3 + \max(B_1, B_2) + \max(B_2, B_3) = A_1 + A_2 + A_3 + B_2 + B_3$
- 二番目に走る人が人 3 の時、
  - 記録は $A_1 + A_2 + A_3 + \max(B_1, B_3) + \max(B_2, B_3) = A_1 + A_2 + A_3 + B_3 + B_3$

よって、記録の最小値は $A_1 + A_2 + A_3 + B_2 + B_3$

小課題2 ($N \leq 3000$, 33点)

タイムBが長い2人を固定した時、
2人よりタイムBが短い中でタイムAが最短の人を選べばよい。

→タイムBでソートした後に前計算しておくことで、定数時間で計算できる。

計算量は $O(N^2)$

小課題2 回答例

```
#include <bits/stdc++.h>
using namespace std;
int main() {
    int N;
    scanf("%d", &N);
    vector<pair<int, int>> X(N);
    for (auto& [a, b] : X) scanf("%d %d", &b, &a);
    sort(X.begin(), X.end());
    vector<int> mini(N);
    mini[0] = X[0].second;
    for (int i = 1; i < N; i++) mini[i] = min(mini[i - 1], X[i].second);
    int ans = 1e9 + 7;
    for (int i = 1; i < N; i++) {
        for (int j = i + 1; j < N; j++) {
            int sum = mini[i - 1] + X[i].first + X[i].second + X[j].first + X[j].second;
            ans = min(ans, sum);
        }
    }
    printf("%d¥n", ans);
}
```

補足

タイム Bの昇順に選手の順番をソート

- pair を使う

(a, b) のペアをソートすると、

- まず a の昇順にソートされ、
- a の値が同じところでは b の昇順にソートされる。

今回は、(タイム B, タイム A)のペアをソートすれば良い。

小課題3 ($A_1 = A_2 = \cdots = A_N$, 10点)

小課題2 で求めた式を使うと、この時の記録の最小値は

$$A \times 3 + B_2 + B_3$$

ただし、$A_1 = A_2 = \cdots = A_N = A$ とした。

よって、選手全員のタイム Bを短い順に $b_1, b_2, \ldots, b_N$ とした時、答えは $A \times 3 + b_2 + b_3$

b_2, b_3 はソートすることで $O(N \log N)$で計算できる。

最小値を求める要領で、$O(N)$ でも計算できる。

小課題4 ($N \leq 200000$, 32点)

3人を選んだ時の記録の最小値は

- $A_1 + A_2 + A_3 + B_2 + B_3 = \mathrm{A}_1 + (\mathrm{A}_2 + \mathrm{B}_2) + (\mathrm{A}_3 + \mathrm{B}_3)$

タイム Bが一番短い人を固定すると、

- 選んだ人よりタイムBが長い人の中で、
- タイムA + タイムB が最も短い 2 人を選べば記録が最短になる。

タイムBでソートした後、適切に前計算をすることで、上の2人は定数時間で計算できる。

- $mi1_{N-1} = \min(A_N + B_N, A_{N-1} + B_{N-1})$, $mi2_{N-1} = \max(A_N + B_N, A_{N-1} + B_{N-1})$
- $mi1_i = \min(mi1_{i+1}, A_i + B_i)$, $mi2_i =$ 「$(mi1_{i+1}, mi2_{i+1}, A_i + B_i)$ の中で2番目に小さい数」

計算量はソートがネックで$O(N\log N)$