

# I: SPORTS DAYS 2.0

---

作題・問題文 汐田(@shioshiota)

テスト 河田(@kawatea03)

# 問題概要

- 重み付き有向グラフが与えられる。
  - 自己辺は含まない。
  - 重みは正
  - 多重辺は含む
- 目的 パスの重みが $k$ 以上となるようなパスの最小の長さを出力せよ。
  - パスの長さが100以下の場合には経路する頂点を順に出力せよ。
  - パスは同じ頂点・辺を何度通ってもよい。
  - 存在しない場合は-1を出力せよ。

# 解法

- 二分法(二分探索)

# 方針

- $K$ がとてもでかい
  - そのせいで、答えの範囲が $1 \sim K$
- $\log K$ にしたい。
- 二分法しよう。

# 前処理

- 前処理として、以下のものを求めます。
- $\text{Cost}[b][i][j] :=$  最大 $2^b$ 本経路したときのノード $i$ からノード $j$ まで移動したときに得られる最大のスコア
- 行列を繰り返し二乗したり、ワーシャルフロイド法を改良したような方法で可能です。
- $\text{Cost}[b][i][j] := \max(\text{Cost}[b][i][j], \text{Cost}[b-1][i][k] + \text{cost}[b-1][k][j])$  ( $0 \leq k < V$ )

# 二分法

- 答えとなる長さをans+1とすると、ansの上のbitから求めるイメージ。
- Solve[i][j] := 途中結果を保存する配列
- ```
for(int b = 20; b >= 0; b--){  
    Int tmp[V][V], flag = true;  
    for(i = 0; i < V; i++)for(j = 0; j < V; j++)for(k = 0; k < V; k++){  
        tmp[i][j] = max(tmp[i][j], solve[i][k] + Cost[b][k][j]);  
        if(tmp[i][j] >= K) flag = false;  
    }  
    If(flag){  
        for(i = 0; i < V; i++)for(j = 0; j < V; j++)solve[i][j] = tmp[i][j];  
        ans += (1 << b);  
    }  
}
```

# 二分法

- 最終的に $\text{solve}[i][j]$ はans本まで経由できるときのiからjまでの移動で実現できる最大のスコアが入ることになる。

# パスの復元

- 今まで紹介した解法にパスを復元できるだけの情報を持たせて頑張ると、一応できます。
- パスの復元が100までしか必要ないことから、小さいケースだけ復元しやすい方法で求めると楽かもしれません。(しなくても解けることは解けます)



# 結果

- First Accept rng\_58 227min
- AC率 50% (4/8)
- 解法
  - wakabaさん, watashiさん、行列累乗風 + 二分法
  - uwiさん ワーシャルフロイド風(紹介したものと一緒) + 二分法
  - rng\_58さん アドホックに

# 元ネタ

- Codeforces Testing Round #4 のB問題と関連があります。  
(アルゴリズム)
- 問題のストーリーは2012年立命館合宿1日目G問題(AOJ 1090 Sports Days)から取ってきています。
- どちらも自分が担当したグラフの(自分の中では)難しい問題だったので同じタイトル同じストーリーにしてみました。
  - 2.0のほうが簡単だったぽい・・・？
- ぜひ併せて解いてみてください。