

JAG 模擬地区予選 2017

I: Revenge of the Endless BFS

原案: hos, oxy

問題文: darsein

解答: cos, darsein, not

解説: darsein

問題概要

- 間違ったBFS (幅優先探索) のコードがある
- このコードを入力の有向グラフで実行した時、探索途中で保持している頂点がグラフの全頂点と一致することがあるか判定せよ
- あるなら、そのために必要な while ループ実行回数も求めよ
- 制約: $2 \leq \text{頂点数} \leq 500, 1 \leq \text{辺数} \leq 2 \times 10^5$

想定解法のポイント

- 見積もり:
不可能ケースでないなら実は答えは N^2+N 以下
- アルゴリズム:
 - 1度すべての頂点を保持した状態になる
 - 以降常に保持した状態になる
 - Yes/No に単調性がある → 二分探索可能
 - 二分探索中では到達可能性行列を繰り返し二乗法 + bit 並列、 $O(N^3/w \log N)$ (w は並列bit数)

想定解法のポイント

- 見積もり:

不可能ケースでないなら実は答えは N^2+N 以下

- アルゴリズム:

- 1度すべての頂点を保持した状態になる

- 以降常に保持した状態になる

- Yes/No に単調性がある → 二分探索可能

- 二分探索中では到達可能性行列を繰り返し二乗法 + bit 並列、 $O(N^3/w \log N)$ (w は並列bit数)

上限見積もり

- 頂点1を含むすべてのサイクルの長さを c_1, \dots, c_k とする
- サイクルがない、あるいはサイクルが互いに素でない ($\gcd(c_1, \dots, c_k) \neq 1$) 場合は不可能
- 素である場合、 i 番目のサイクルを x_i 回通るとすると、 $c_1x_1 + \dots + c_kx_k$ step目に頂点1にすることが可能
- F 以上の値をすべて $c_1x_1 + \dots + c_kx_k$ の形で表現できるような F が存在、 $F + n$ step あれば全頂点保持できる

フロベニウス数

- フロベニウス数 F : $c_1x_1 + \dots + c_kx_k = F$ が解を持たなくなる最大の F
- F は $\min c_i \times \max c_i$ で抑えられる
 - $c_1 = \min c_i, c_k = \max c_i$ とする
 - フロベニウス数は、 c_1 頂点で u から $(u + c_i) \bmod c_1$ にコスト c_i の辺を貼ったグラフで最短路を解くことで得られる
 - 0 から v への最短路を d_v として、 $F = \max d_v$
 - 最短路に含まれる1辺の長さ $\leq c_k$ 、辺数 $\leq c_1 - 1$ より、最短路は $c_1 c_k$ 以下
- サイクル長 $c_i \leq N$ より、 $F \leq N^2$

想定解法のポイント

- 見積もり:
不可能ケースでないなら実は答えは N^2+N 以下
- アルゴリズム:
 - 1度すべての頂点を保持した状態になる
 - 以降常に保持した状態になる
 - Yes/No に単調性がある → 二分探索可能
 - 二分探索中では到達可能性行列を繰り返し二乗法 + bit 並列、 $O(N^3/w \log N)$ (w は並列bit数)

想定解法: 繰り返し2乗法+二分探索

- グラフの隣接行列 = 1 stepでの到達可能性行列
- 隣接行列のk乗 = k stepでの到達可能性行列
- 繰り返し2乗法で k step目を計算: $O(N^3 \log k)$
- 1~F step を二分探索
- $O(N^3 \log F)$ の計算を $O(\log F)$ 回するので
 $O(N^3 \log^2 F) \rightarrow F = O(N^2)$ より $O(N^3 \log^2 N)$
- BFSを $O(N^2)$ Step 計算: $O(N^2 (N+M))$ より速い

想定解法: 繰り返し2乗法+二分探索

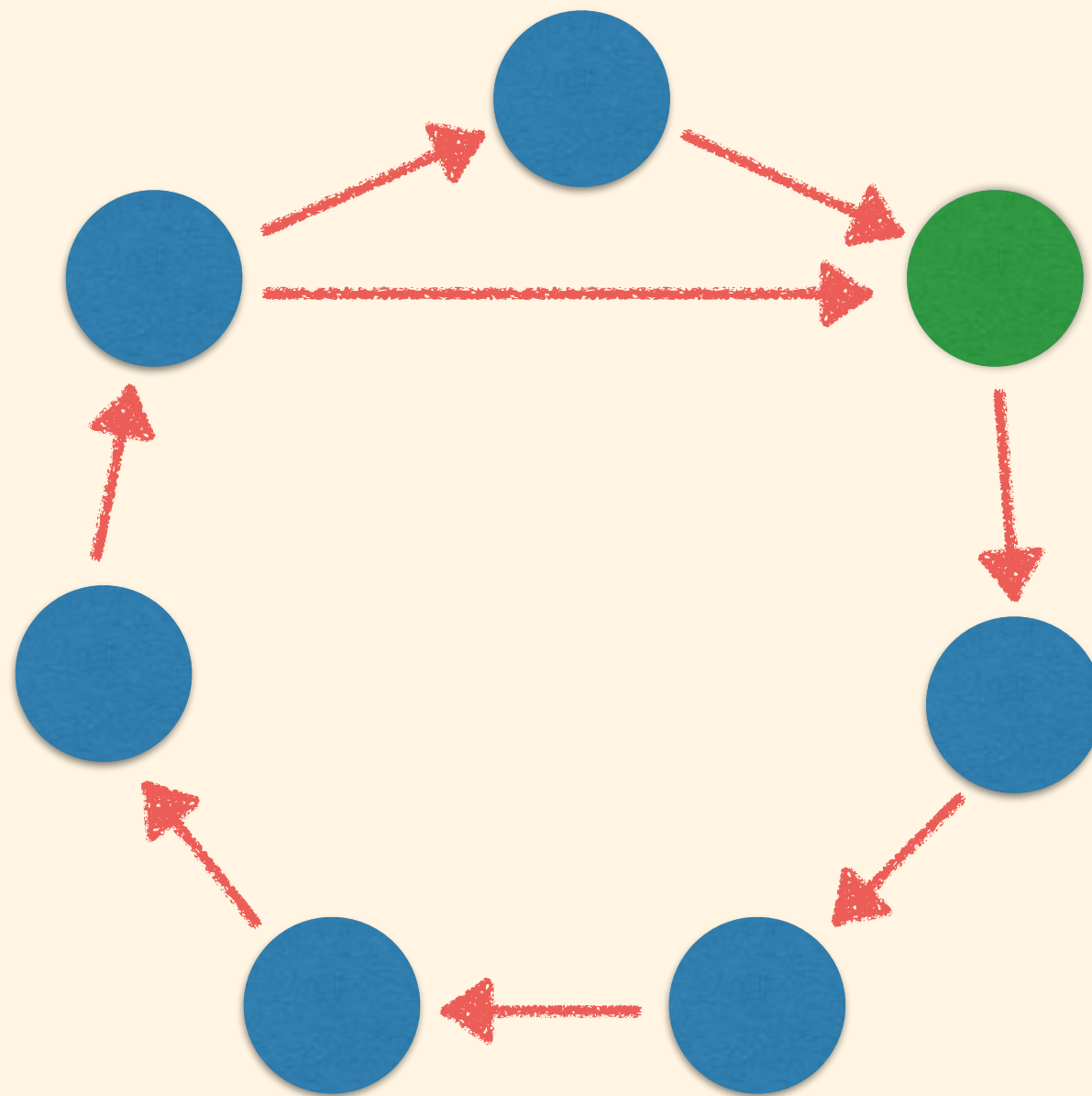
- 答えの上位bitから決めて行く方法で \log が1つ落とせる
- あらかじめ行列の 2^i 乗を計算
- 上位bitから現状の答えに掛け合わせてみる
- 掛けた結果が Yes なら行き過ぎているかもしれないので戻す、No なら掛ける
- 最終結果は No となる最大値なので、+1が答え

想定解法: bit 並列行列乗算

- 普通の行列累乗: $O(N^3)$
- 今回は true/false だけからなる行列なので計算をbit並列できる: word長 w として $O(N^3/w)$
- 行列の各行をwordにパックする
- $C = A \times B$ の計算時は A と B の転置 (B^t)を用意
- $C_{i,j}$ の計算: A_i と B_j^t の論理積 (AND) を計算して 1bit でも true なら $C_{i,j} = \text{true}$

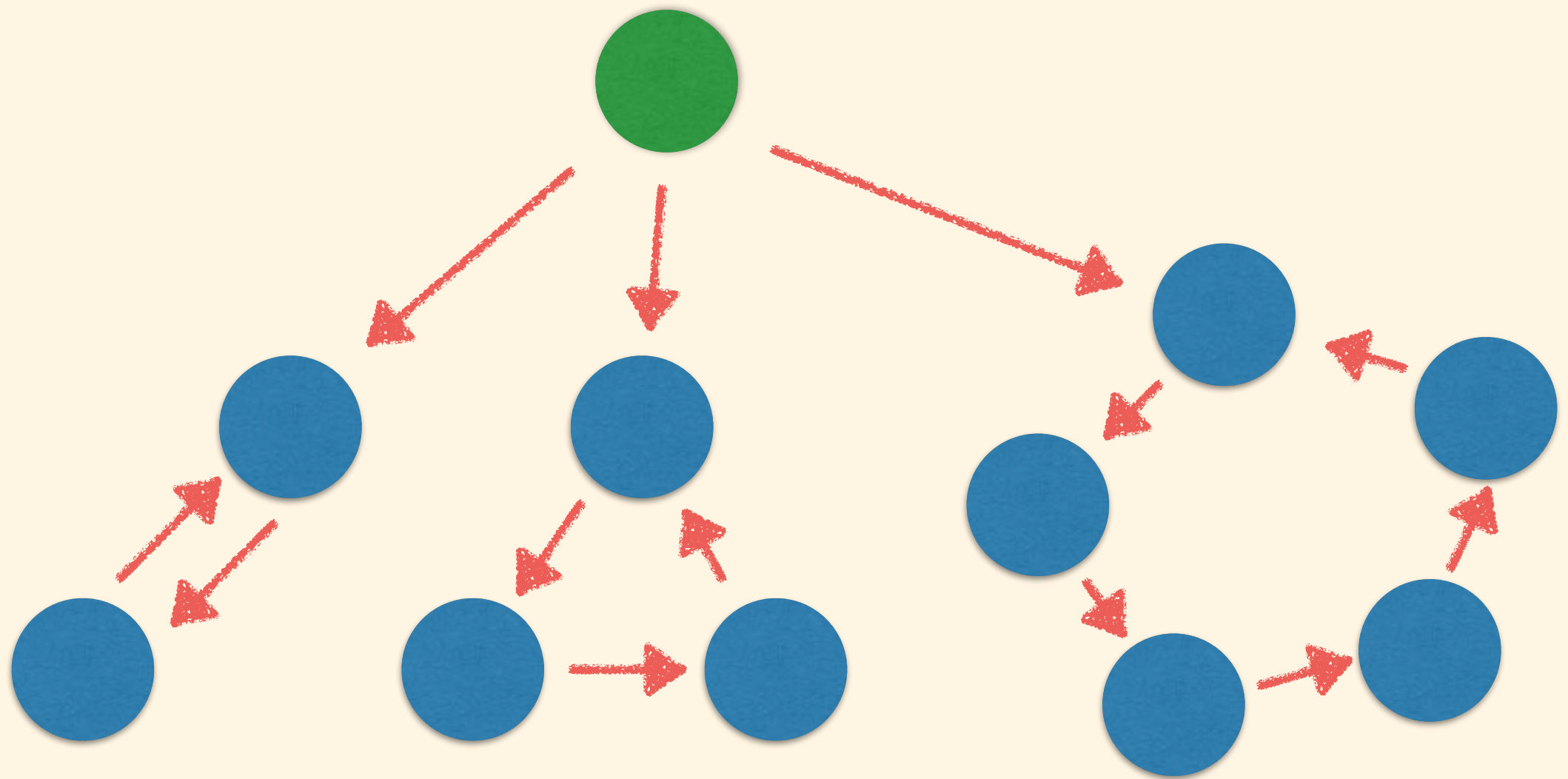
テストケース

- 答えが N^2-N



テストケース

- 状態がループするまでに指数時間かかる



Writer 解

- cos: 55 行 1273 bytes (C++)
- darsein: 128 行 2617 bytes (C++)
- not: 49 行 1027 bytes (C++)

統計情報

- AC / submissions
 - 7 / 26 (27%)
- First Acceptance
 - CxivDxiv (49:09)