いけるかな？

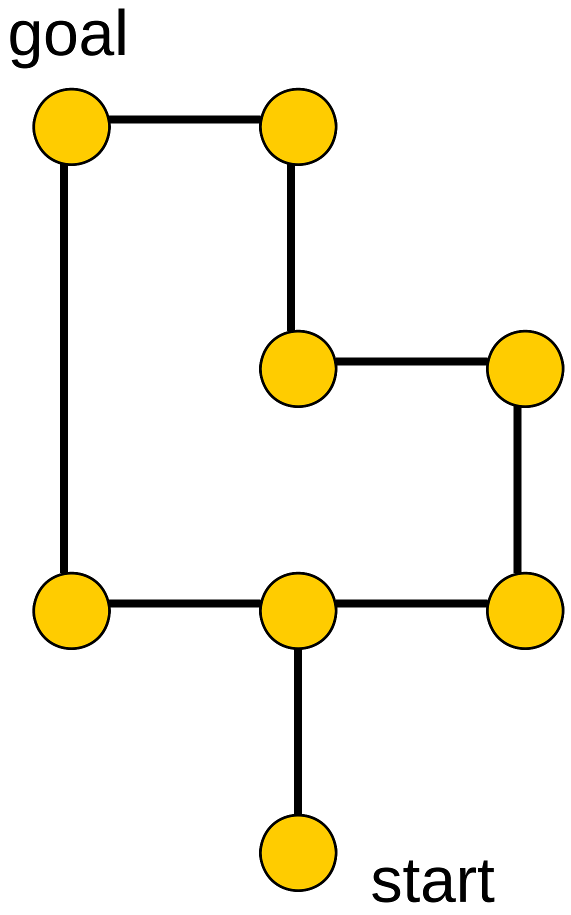
問題作成、解説担当：中島

副担当：坪坂、松本

問題の概要

- 無向グラフが与えられる
- ある頂点から他の頂点まで決められたステップ数で行けるか
- ただし、後戻りすることはできない
- 頂点数、枝数は50まで
- ステップ数は2^31まで

具体例



ステップ数:10

前処理

- 後戻り不可能条件に対応させるため、「現在グラフ上のどこにいるか」を、(現在位置,1ステップ前にいた位置)で表現する。
- 状態数は枝の数の2倍になる
- この状態1つ1つを頂点とし、1ステップでの遷移を枝としたグラフを構成する

駄目な回答例(多くの場合では有効な方法)

- グラフをステップ数倍に拡大した上でのグラフ探索
  - 各頂点を(頂点番号, 残りステップ数)に拡大
  - (スタート地点, 初期ステップ数)から(ゴール地点, 0)まで到達可能か
  - 幅優先探索をすることで空間計算量はO(M)にはなる
- ステップ数が非常に大きいためこれでは制限時間内に正解することは不可能！

解法

- 行列Akの各要素ak_ijを
  - 頂点iから頂点jへkステップで到達可能な場合:1
  - 到達不可能な場合:0

として定義すると、Akは以下の性質がある

  - A1は元のグラフの隣接行列
  - AnとAmからA(m+n)が計算可能
- よって、行列乗算と同様の方法で、行列の計算回数をlog(Z)回に落とすことができる

知識1

- 行列Aの各要素a_ijを
  - 頂点iから頂点jへ接続しているの場合:1
  - 接続してしていない場合:0

としたとき、A^Zのij成分がiからjへZステップで移動するための場合の数

知識2

- (N×N)の行列AのZ乗はO(N^3*log(Z))で計算することが可能
- matrix pow(matrix A, int Z){

```
 if(Z == 1)return A;
 matrix B = pow(A, Z/2);
 if(Z % 2 == 0){
     return B*B; // O(N^3)
 } else {
     return B*B*A; // O(N^3)
 }
}
```

別解

- 以下の方法でもジャッジデータに対して正解可能
  - 行列Akを毎ステップシミュレーションしてkの値を1つずつふやしながら計算する
  - iステップ目の行列Aiがjステップ目(j<i)の行列と同じになったら、残りステップ数を(j-i)で割った余りにする
  - 以上の2つを残りステップが0になるまで繰り返す
- これで解けそうな理由（完全な考察はしていません）
  - kを増やしていくと最終的にAkは周期的な動きになりそう
  - 収束までのステップ数はあまり大きくなさそう
    - 1500ステップ以上計算するようなデータを作ることができませんでした

回答状況

- Submitted:3
- Accepted:1/2
- 1st Acceptance:Takahashi Shuuhei

コメント

- 少し知識色の強い問題
- しかし、知識2くらいは知っている人が多いはず
  - 知らなかった人はぜひ覚えましょう
- 想定解法以外でもジャッジデータを通すことは可能
  - 問題に対して柔軟なアプローチを
  - 解法は1つとは限りません

最後に

- これでサイコロが1億個になっても大丈夫!