

MinimumCostPath

JAG Summer Camp 2012 Day3-B: Problem F

2012/09/16

原案: 薮

解答: 薮, 大坂

解説: 大坂

問題概要

- ▶ $N \times N$ の方眼に $M (\leq 50)$ 個の障害物が置かれている
- ▶ $(1,1)$ から (N, N) に最短距離で移動する方法は何通りあるか

解法

- ▶ 包除原理+DP+BFS
- ▶ $N \leq 1000$
 - BFSで求める(余裕)
- ▶ $N > 1000$
 - BFSではTLE(絶望)

N が大きい時

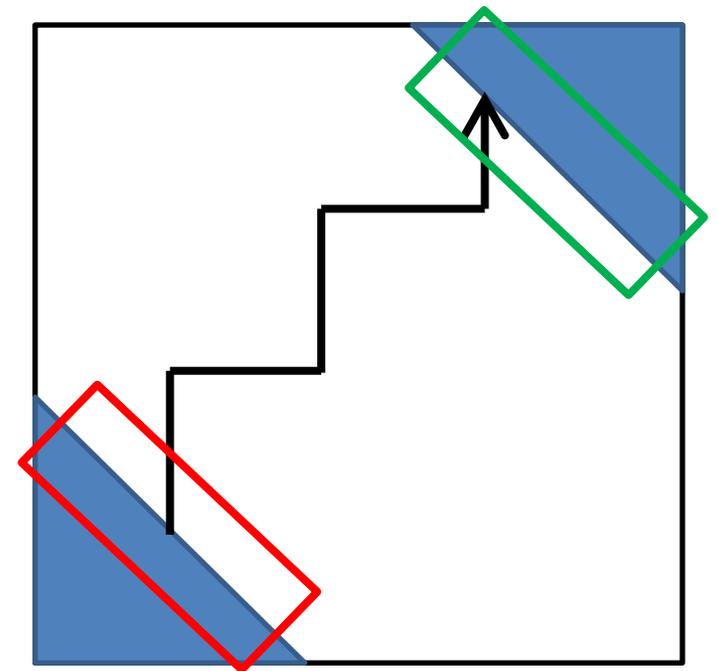
- ▶ 最短経路長が $2(N - 1)$ とすると...
 - \Leftrightarrow x座標, y座標は必ず増加する
 - ▶ 障害物を無視した経路数は二項係数で計算できる
 - ▶ 障害物を1つも通らない $(1,1)$ から (N, N) への経路数
- ↑ 包除原理を使えばもとめられる
- $O(2^M)$ なので間に合わない \rightarrow DPで $O(M^2)$

経路数のDP

- ▶ $dp[P]$: P 以外の障害物を通らない始点から P への経路数
- ▶ $dp[P] =$
障害物を無視した始点から P への経路数
− $\sum_{Q:Pより前} dp[Q] \times (QからPへの経路数)$
- ▶ DPは座標でソートした順に行う

うまくいくためには

- ▶ $(1,1)$, (N,N) からBFSする
 - 距離 $2M$ 位
- ▶ (始点, 終点)を結ぶ最短経路で
 x 座標, y 座標が必ず増加するものがある
- ▶ 赤から緑への最短経路は
 x 座標, y 座標が必ず増加
- ▶ $(1,1)$ から赤までの経路数 (BFS)
- ▶ 赤から緑までの経路数 (DP)
- ▶ 緑から (N,N) までの経路数 (BFS)
をそれぞれ求める



計算量

- ▶ 二項係数のための階乗の前計算: $O(n \log n)$
- ▶ (1,1)から赤までの経路数: $O(M^2)$
- ▶ 赤から緑までの経路数: $O(M^2)$
- ▶ 緑から(N, N)までの経路数: $O(M^2)$

- ▶ 総計算量: $O(n \log n + M^2)$

ジャッジ解

- ▶ 藪 C++ (192行)
- ▶ 大坂 Java (243行)

解答状況

- ▶ First Acceptance
 - Onsite: agitsune (95 min)
 - All: agitsune (95 min)
- ▶ First Submission
 - Onsite: agitsune (95 min)
 - All: agitsune (95 min)
- ▶ AC/Submission: 1/2
- ▶ AC rate: 50%