

「盗まれた宝石」の解説

原案：八森

模範解答作成：小西、八森

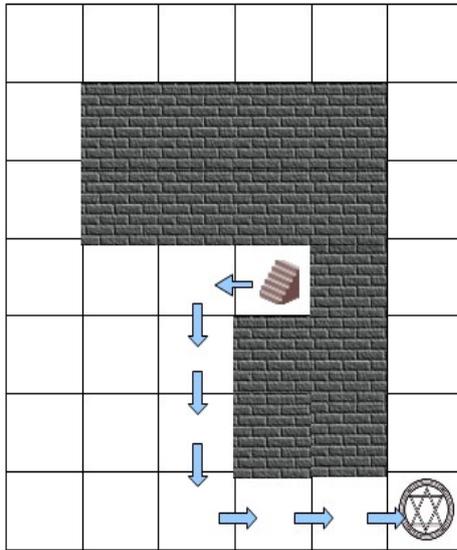
解説：八森

問題概要 (1/2)

- 迷路と禁止パターンが与えられる。禁止パターンを取らずに、スタートからゴールへ移動するのに必要な移動パターン長の最小値を求めよ。
 - 迷路のサイズ： $1 \leq \text{縦サイズ}$ ， $\text{横サイズ} \leq 50$
 - 禁止パターン数： 10 以下
 - 禁止パターン長さ： 10 以下

問題概要 (2/2)

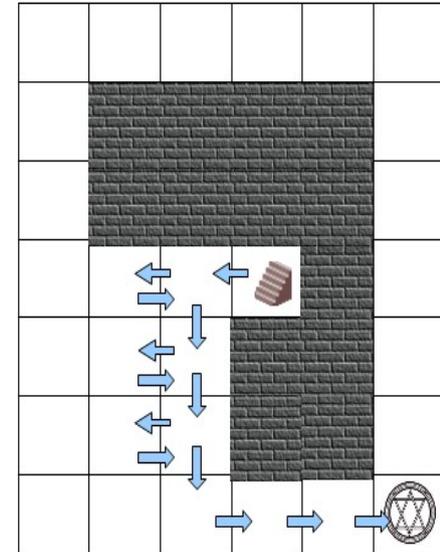
- 駄目な例
(禁止パターンが移動パターンに含まれる)



移動パターン: ← ↓ ↓ → → →

禁止パターン: { ← ↓
↓ ↓
← ← ←

- 良い例
(長さが最小で、禁止パターンを含まない)



移動パターン: ← ← → ↓ ← → ↓ ← → ↓ → → →

禁止パターン: { ← ↓
↓ ↓
← ← ←

基本方針

- 現在位置と、それまでにどのような移動パターンをとったかを状態とした **BFS** 。
 - 移動先に壁があったらその移動は行わない。
 - それまでの移動パターン + 次に移動する方向 {↑ or → or ↓ or ←} が禁止パターンに含まれていたらその移動は行わない。
 - その中で、
状態 (スタート地点, 移動無し) から
状態 (ゴール地点, 任意の移動パターン) に
遷移するのにかかる最小回数を計算。

駄目な解法①

- 現在位置、移動パターンの最後 10 回分を状態とした **BFS**
 - 禁止パターン長さが最大で 10 なので、移動パターンの最後 10 回分のみ記憶すれば、それを参照して禁止パターンを取らないような移動ができる。
- ですが、状態数が爆発してしまいます。
 - 現在位置 : $50*50$
 - 過去 10 回分の移動パターン : 4^{10}
 - $50*50*4^{10} \doteq 26$ 億

状態数を減らすために

- 考え方
 - 禁止パターン数、禁止パターン長が小さいところに注目！
 - 禁止パターン数：10 以下
 - 禁止パターン長さ：10 以下
 - (それまで禁止パターンをとらなかつた) 現在までの移動パターンが禁止パターンに含まれてはならない！
 - 禁止パターンのマッチング状況を状態として持たせる。
 - マッチング状況がわかると移動パターンが復元できる。
 - 状態 (現在位置、移動パターンの最新何文字分が
どの禁止パターンの先頭の一部となるか)
で **BFS** すると良い！

禁止パターンとのマッチング状況の例

マッチング状況：禁止パターン1の先頭4文字目まで一致。
(移動パターンは    と復元される。)

禁止パターン：
    
  
     

    のつぎに  へ行く時：新しい移動パターン      は禁止パターン    を含んでしまう。

    のつぎに  へ行く時：新しい移動パターン      は禁止パターン3の先頭5文字目まで一致する、
というマッチング状況の状態ができる。

駄目な解法②

- 状態 (現在位置、禁止パターンの ID, 移動パターンとの一致文字数) で BFS。
- 現在の状態と次の移動方向を元に新しい状態に遷移するとき、移動パターン最新 x 文字分と全禁止パターンとのマッチングを調べる。
 - 逐次マッチングをしていたら、計算量が大きくなってしまいます。
 - 状態数 : $O(50*50*10*10)$
 - マッチングのオーダー : $O(4*10*10)$
 - 移動方向 = 4 通り , 禁止パターン数 = 10 、パターン長 = 10
 - $O(50*50*10*10)*O(4*10*10) = O(1 \text{ 億})$

高速化方法 (by 小西)

- 前処理で (禁止パターンとのマッチング状況 + 次の移動方向) に対する新しいマッチング状況を作ることで、逐次マッチングを防ぐことができる!

- 作るテーブル:

tbl[禁止パターンとのマッチング状況][移動方向] := 新しいマッチング状況

- 計算時間

- 禁止パターンとのマッチング状況数 : $O(10*10)$
- 移動方向 : $O(4)$
- 新しいマッチング状況の候補数 : $O(10*10)$
- 新しいマッチング状況となりうるかの判定 : $O(10)$
- $O(10*10*4*10*10*10) = O(40 \text{ 万})$

想定解法

- 状態 (現在位置、禁止パターンの ID, 移動パターンとの一致文字数) で BFS。
- 現在の状態と次の移動方向を元に新しい状態に遷移するとき、前述したテーブルを参照して新しい状態に遷移。
- 計算時間
 - 前処理時間 : $O(40 \text{ 万})$
 - 状態数 : $O(50*50*10*10)$
 - 計算時間 : $O(40 \text{ 万}) + O(50*50*10*10)*O(4 \text{ 方向}) = O(140 \text{ 万})$