



CatAndRabbit

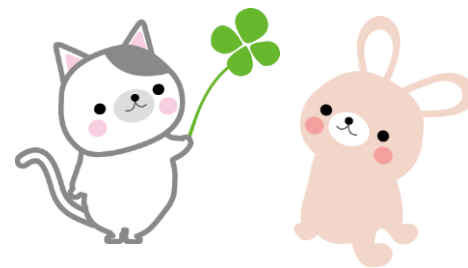
TopCoder SRM558 D2-Hard

writer: lyrically



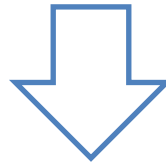
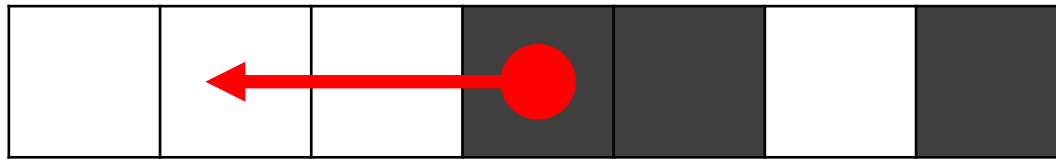
問題概要

- Cat (先手) と Rabbit (後手) でゲーム
- 1 列に並んだ白黒のタイル
- 各ターンの行動：
 1. 黒いタイルに乗る
 2. 隣の白いタイルに何回か (1 回以上) 動く
 - タイルは乗った瞬間黒くなる
- 行動できなくなったら負け
- どっちが勝つか？



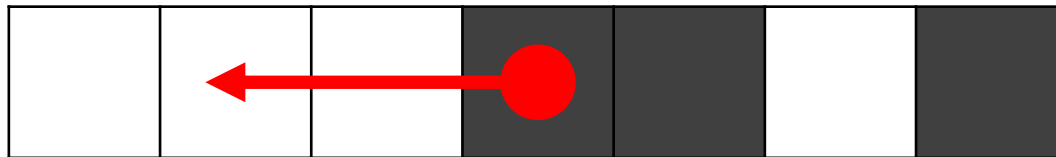
問題概要

- 行動の例



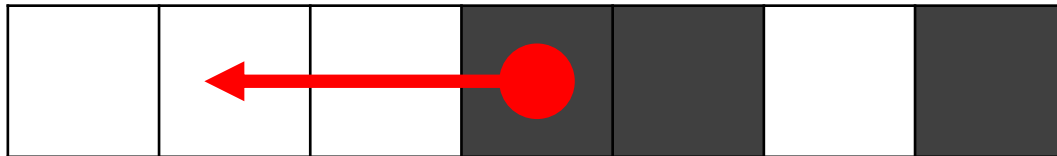
考察

- 白いタイルに隣接した黒いタイルからスタートしなければならない
- 「白いタイルが連続した部分」が左端または右端から削られていく
 - 真ん中あたりから分断されることはない



考察

- ということは…？



考察

- 石取りゲームっぽい！
- 「白いタイルが連続した部分」が石の山に対応



1

2

4

考察

- 石取りゲーム
 - 石の山がいくつがある
 - 各ターンの行動は, ある山から 1 個以上の石をとること
 - 行動できなかつたら (石がなかつたら) 負け
- これは "Nim" として知られている

Nim

- 最初の山の石の個数を a_1, \dots, a_n とする
- 先手が勝つ必要十分条件は,

$$a_1 \text{ xor } \dots \text{ xor } a_n \neq 0$$

- xor : 二進法で各桁ごとに足し算

– ただし $1 + 1 = 0$

– 例 : $12 \text{ xor } 5 = 9$

- 二進法で書くと $1100 \text{ xor } 0101 = 1001$

Nim

- 次の 2 つを証明すればよい
 - xor が 0 でない値ならば, うまく石をとって 0 にできる
 - xor が 0 ならば, どう石をとっても 0 でない値にしかできない
- 今回は詳しい証明は省略
 - Web 等に資料がたくさんあります

コーナーケース

- 本当に石取りゲームと同じ??
- すべてのタイルが白いとき, ルールに従うと何も行動できない (Example #4)
 - 後手の Rabbit の勝ち
- この場合以外は「白いタイルが連続した部分」の左または右が黒いので大丈夫

解法まとめ

- 全部のマスが白だったら "Rabbit" を返す
- 「白いタイルが連続した部分」のそれぞれの長さを求める
- それらの xor を計算し, 0 でないならば "Cat" を, 0 ならば "Rabbit" を返す

解答例 (Java)

```
public String getWinner(String tiles) {  
    if (tiles.matches("[.]*")) {  
        return "Rabbit";  
    }  
    int x = 0;  
    for (String part: tiles.split("#")) {  
        x ^= part.length();  
    }  
    return (x != 0) ? "Cat" : "Rabbit";  
}
```

おまけ

- 「行動できなくなったら勝ち」とするとどうなるでしょう？
- 「黒いタイルを選ぶ→何回か動く」を1ターンに1回以上 k 回以下できる, とするとどうなるでしょう？
- 上の両方を取り入れるとどうなるでしょう？
 - 考えてみてください