



Karakuri Doll

問題原案：三廻部

解説・データ作成：林崎

解いた人：寺島、福澤、牟田



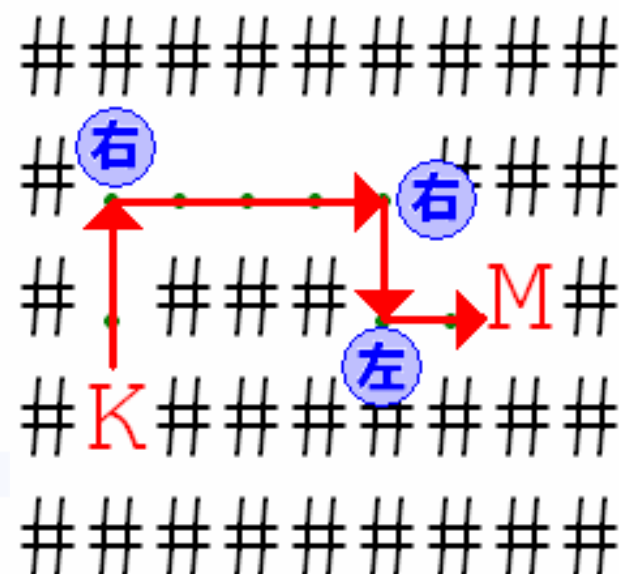
問題 (1)

- 右下のようなマップが与えられます
- #が壁、.が床
- 人形がK(Kitchen)に居ます
- M(Master)まで行き、
またKに戻りたい
K, Mの三方は壁(#)

```
#####  
#.....###  
#.###.M#  
#K#####  
#####
```

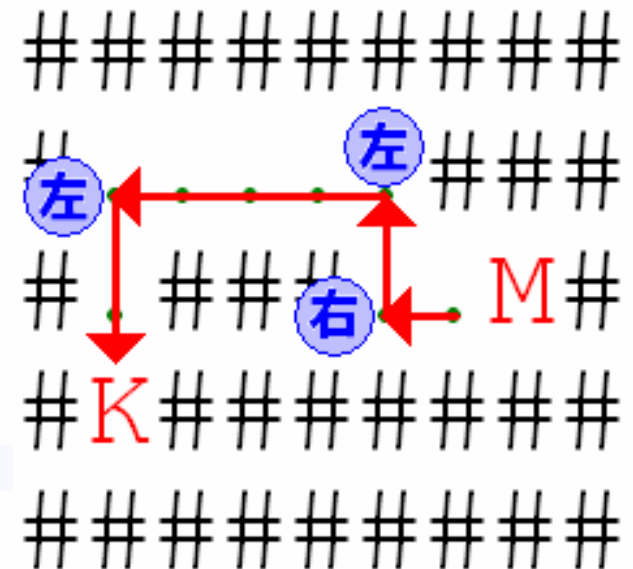
問題 (2)

- 人形には「左」「右」からなる有限のプログラムを与えます
- Kを出発し、壁にぶつかるまで直進し、ぶつかったらプログラムに従って左か右に曲がります
ぶつかるまで曲がりません
- 右の例: 「右右左」とプログラムするとMに辿りつきます



問題 (3)

- 帰りは、プログラムを「逆実行」します
逆順、かつ左右反転の曲がり方をする
行き: 右右左
帰り: 右左左
この例では往復可能



問題 (5)

- 以下のどれかを出力してください

K M K と往復できる

- 往路復路ともにプログラムが尽きるまで動作を続け、その時点でぴったりM/Kに居ればOK
- 例えば復路途中でKに一度帰還しても、最終的にKに居なければ不可

K M は行けるが往復はできない

K M も行けない

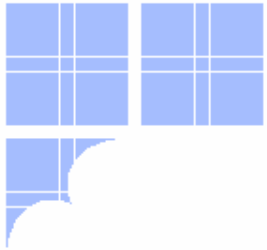
問題 (6)

- 細かい条件が多いです
 - 途中でKやMを通過しても良い
 - 最後はぴったりM/Kで止まる必要がある
 - 最後に止まった時にM/Kにいれば
どっちを向いていてもよい
 - K,M出発時は唯一壁のない方向を向いている



Sample Input 最後

- さてここで陰険データ解説。



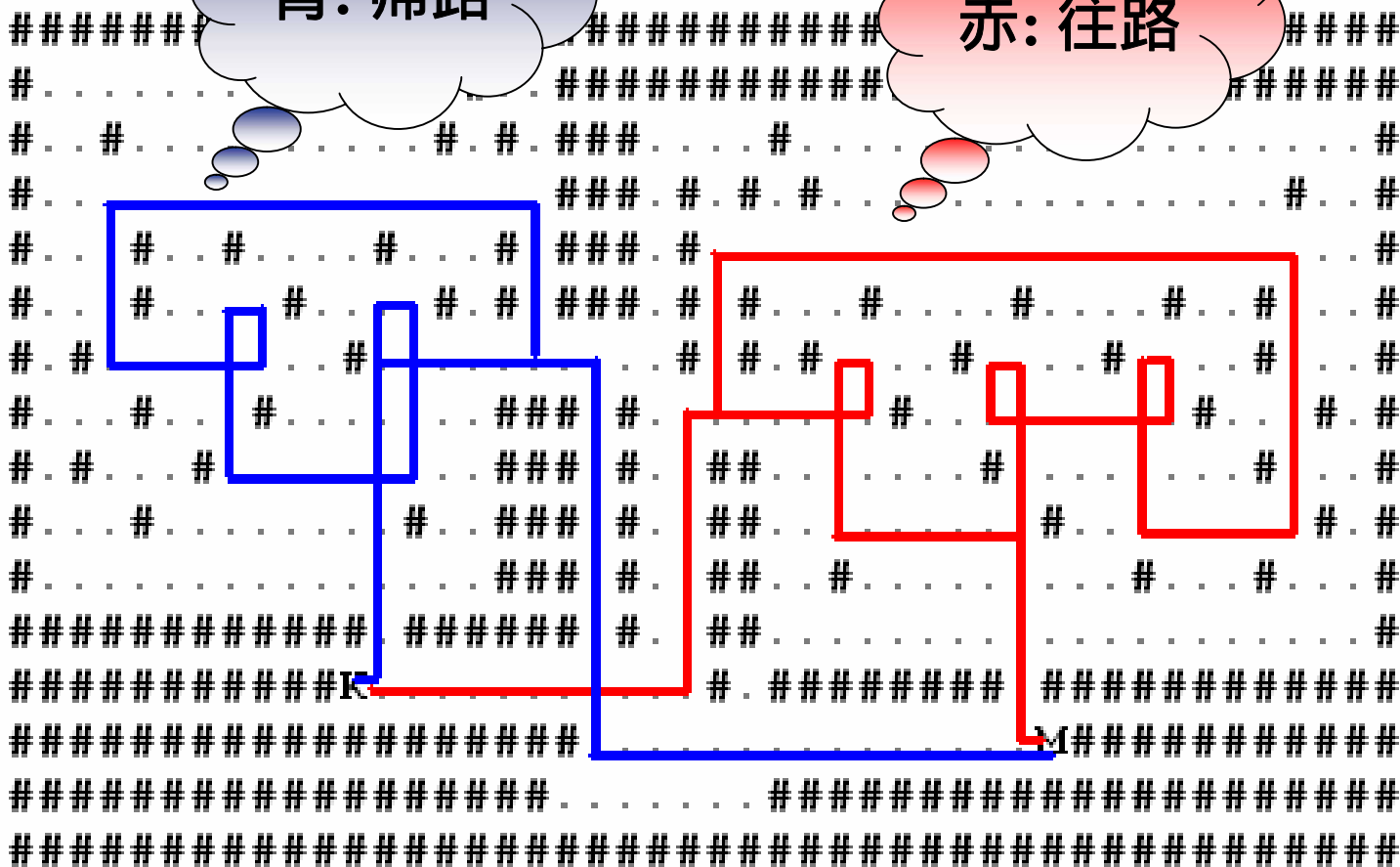
Sample Input 最後

```
#####  
#.....#...#####  
#.#.....#.#.###.....#  
#.....###.#.#.#.....#.#  
#.#.#.#.#.#.###.#.....#  
#.#.#.#.#.#.###.#.#.#.#.#.#  
#.#.#.#.#.#.###.#.#.#.#.#.#  
#.#.#.#.#.#.###.#.#.#.#.#.#  
#.#.#.#.#.#.###.#.#.#.#.#.#  
#####.#####.#.###.....#  
#####K.....#.#####.#####  
#####.....M#####  
#####.#####  
#####
```

Sample Input 最後

青: 帰路

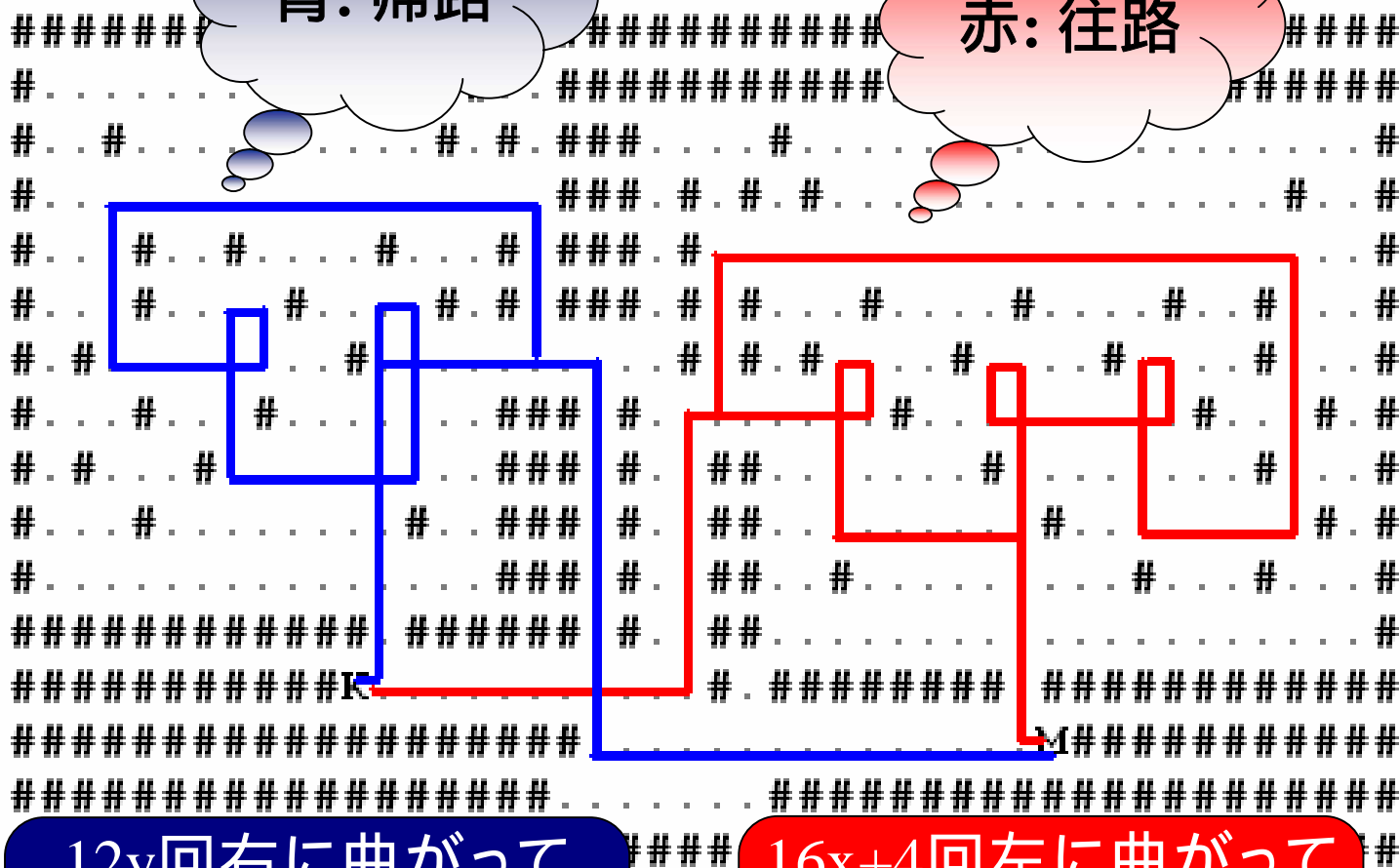
赤: 往路



Sample Input 最後

青: 帰路

赤: 往路



12y回右に曲がって
1回左に曲がると
抜けるループ

16x+4回左に曲がって
1回右に曲がると
抜けるループ



Sample Input 最後

- 同じ所をぐるぐる回ることがあります
- プログラムの長さは40
左右(左*36回)右左

解法 (1)

- 片道だけなら、(位置,向き)をノードとする
拡大グラフ上で到達可能判定

最初に出発すると、
(最初に壁にぶつかる位置、
その時の向き)
に到達する

- 右の赤矢印(K'とする)

そこから左or右に曲がって壁に
ぶつかった時の(位置,向き)にも到達可能

```
#####  
#↑...###  
#.###.M#  
#K#####  
#####
```

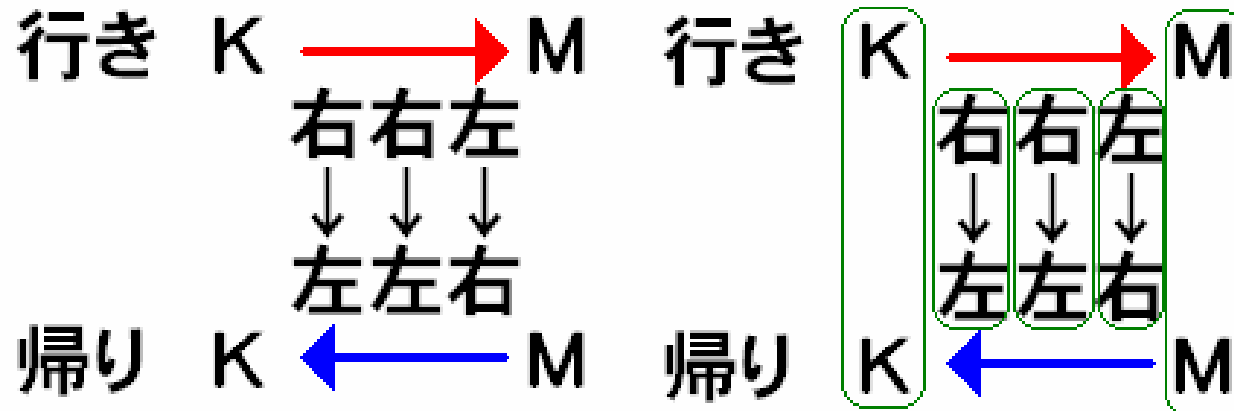
解法 (2)

- 往復: 行きと帰りをあわせて考える

片道の状態は(位置、向き)

曲がり方もまとめられる

(K', K) から (M, M') まで行けるか



解法 (3)

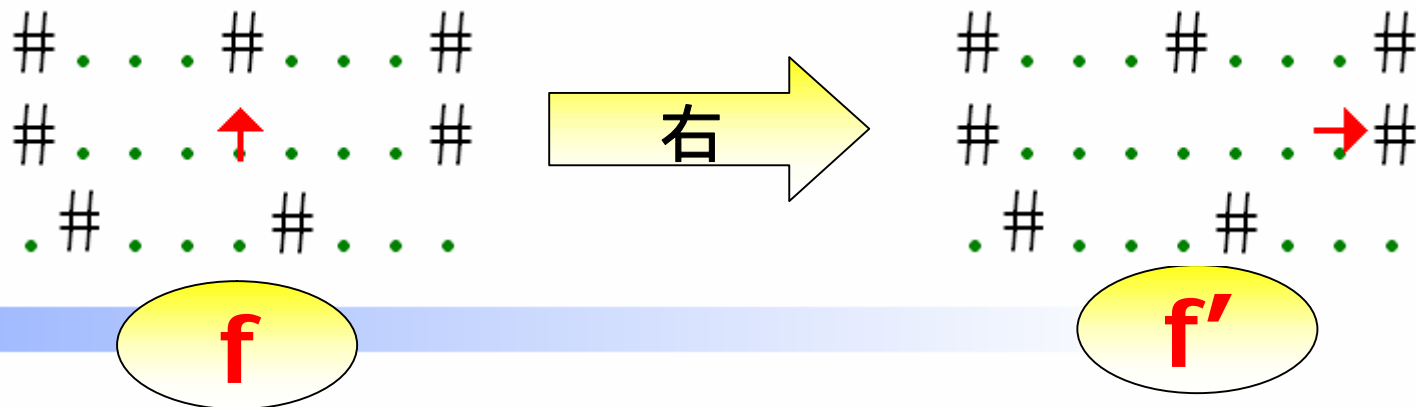
- 開始状態: (K' , K)
右上図参照
復路状態は向き任意
表現の厳密性にやや
欠けるがご了承ください
- 目標状態: (M , M')
往路は向き任意

```
#####  
#↑.....###  
#.###.M#  
#↓#####  
#####
```

```
#####  
#.....###  
#.###←.M→#  
#K#####  
#####
```

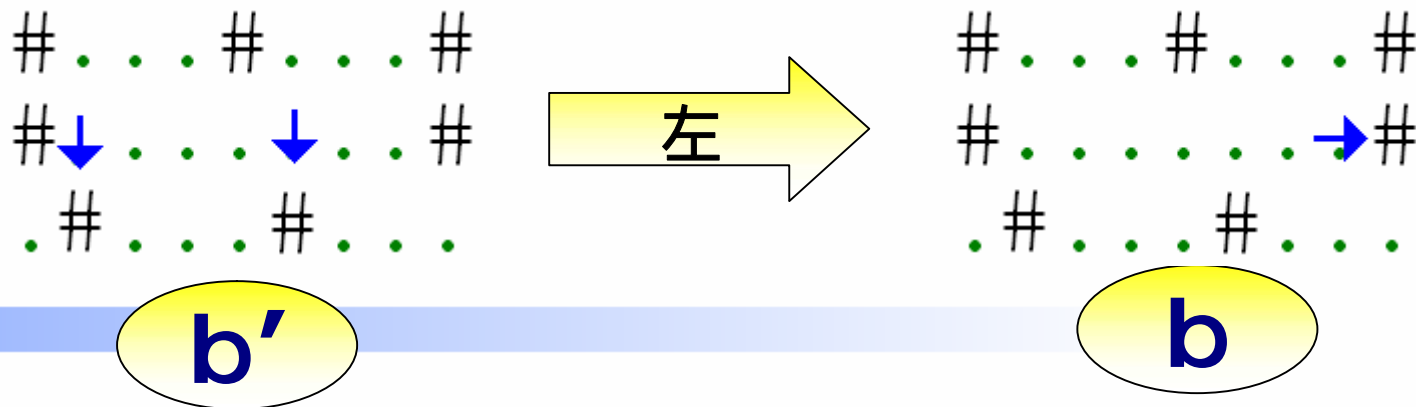

解法 (4)

- ステート(f, b)から、
右(行きの場合)に曲がるとどうなるか?
行きに関しては、普通に f から**右**に曲がり、
壁にぶつかるまで直進して辿りつくステート f'
を計算



解法 (5)

- ステート(f, b)から、
右(行きの場合)に曲がるとどうなるか?
帰りに関しては、「 b' から**左**に曲がって直進したら b になる」ような b' の集合を求める
各 b' について、 (f, b) から (f', b') まで到達可能



解法 (6)

- 計算量 = ステート数
= マスの数 n に関して $(4n)^2$
片道ステート数が 場所の数 * 4方向
 $64 * 16 = 1024$ マスなので16M
実際にはもっと少ない
- ダメな解法例
片道の経路を全列挙して戻れるかを検証:
経路長長い&&ループが存在するので無理
片道ごとの到達可能判定



解法 (7)

- 出題者側による解答

4人がそれぞれ作成

- C++ 2人、Java 2人

処理時間: 各入力 2 ~ 25秒 (CoreDuo 1.83GHz)

行数: 100 ~ 250行





Submission Statistics

- 5 Submit (5チーム) 2 Accept

First Accept: _____ (165分)

Second Accept: Makegumi (175分)

- いずれも想定解法と同じ

解説担当の事前予想は6 Submit 2 Accept

Accept したチームはおめでとうございます

- Judge roomが沸きました

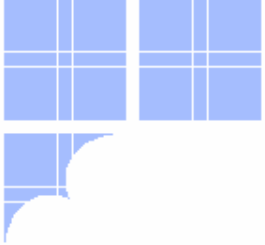


講評 (1)

- アルゴリズムの発想力と実装力とが必要
解法を思いつけるか
 - 知識問題ではない
- 実装量が多い
 - 自分で書いて結構面倒だった
 - ライブラリを使う問題ではない
- 難度としてはアジア予選でも自然なレベル

講評 (2)

- 今回は優しいSample Inputにしました
問題の性質上かなりバグりやすい
 - 終端での条件を間違えると即死想定誤答例はSample Inputで全部落とすようにしたのでSample InputでデバッグすればOK
普通はこんなに優しくありません



おまけ：テストデータ例（1）

- Sample Input最後と同様のループ系
自動生成器を作った
 - 作らないとやってらんない $16x = 16y + 4$ みたいな往復不可能なものも
- ランダム生成(120万データセット)
誤答しやすいものや往復するためのプログラムが長い物を拾い出す
想定誤答例を作成

おまけ：テストデータ例（2）

- プログラム最長：80
左右(左*72回)右左

- 56 16

```
#####  
#.....#..#####  
#..#.....#.#.###...#.....#  
#.....###.#.#.#.....#..#  
#...#..#...#...#...#...#...#...#...#...#...#  
#...#...#...#...#...#...#...#...#...#...#...#  
#.#.....#.....#.....#.#.#.....#.....#...#  
#...#...#.....#.....###.#.....#.....#...#  
#.#...#.....#.....###.#...##.....#.....#...#  
#...#.....#.....#...###.#...##.....#.....#...#  
#.....###.#...##.#.....#...#...#  
###.#####.#...##.....#  
#K.....#..#####.###  
#####.....M#  
#####.....#####  
#####
```

おまけ：テストデータ例（3）

■ ランダム生成中プログラム最長：69

左右左左左左右左左左右左左左右右右左左右左左右左左右右右左左左右右右左右
右右左左右左右右右左右右左右右左左左右右右左右右右左左右右右右

13 15

```
#####  
#.....#...#  
#...#...#...###  
#..#M#..#.#.#  
##.#....K#.#.#  
#.....#...#  
##.##.....##  
#.....##  
##....###.#.#  
#.###...#...##  
#.#...#.....#  
#..#...#.#...##  
#...#...#.#  
#.###...#...#  
#####
```