

TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI



BÁO CÁO TỔNG KẾT

BÀI TẬP LỚN MÔN THUẬT TOÁN VÀ ỨNG DỤNG

BÀI TOÁN: Errich-Tac-Toe

Sinh viên thực hiện:

Họ và tên: Lê Quang Duy

Mã sinh viên: 181203460

Lớp: CNTT1 - K59

Khoa: Công nghệ thông tin

Người hướng dẫn: ThS Phạm Xuân Tích

Hà Nội 2020

This image shows a full page of primary-ruled paper. It features approximately 20 horizontal rows, each defined by two parallel dotted lines. The lines are evenly spaced and extend across the entire width of the page, providing a guide for handwriting practice. There are no margins, text, or other markings on the paper.

Giảng viên

Mục lục

I. Phát biểu bài toán:	3
1.1 Đầu vào:	3
1.2 Đầu ra:	3
1.3 Miền ràng buộc:	3
Ví dụ:	4
II. Nêu ý tưởng:	5
III. Minh họa thuật toán:	8
IV. Mã nguồn:	9
V. Tính toán độ phức tạp:	10

Hình vẽ minh họa:

Figure 1Minh họa trò chơi Tic tac toe	3
---------------------------------------	---

I. Phát biểu bài toán:

Errich đã đưa ra một thử thách và đố bạn của mình giải được.

Trong trò chơi Tic-Tac-Toe, gồm một ma trận ô vuông có n hàng và n cột. Mỗi ô có thể trống (.) hoặc chứa ký tự (X) hoặc (O). Nếu tồn tại 3 ký tự giống nhau liên tiếp theo hàng hoặc cột, người chơi sẽ chiến thắng, nếu không thì họ đang hòa.

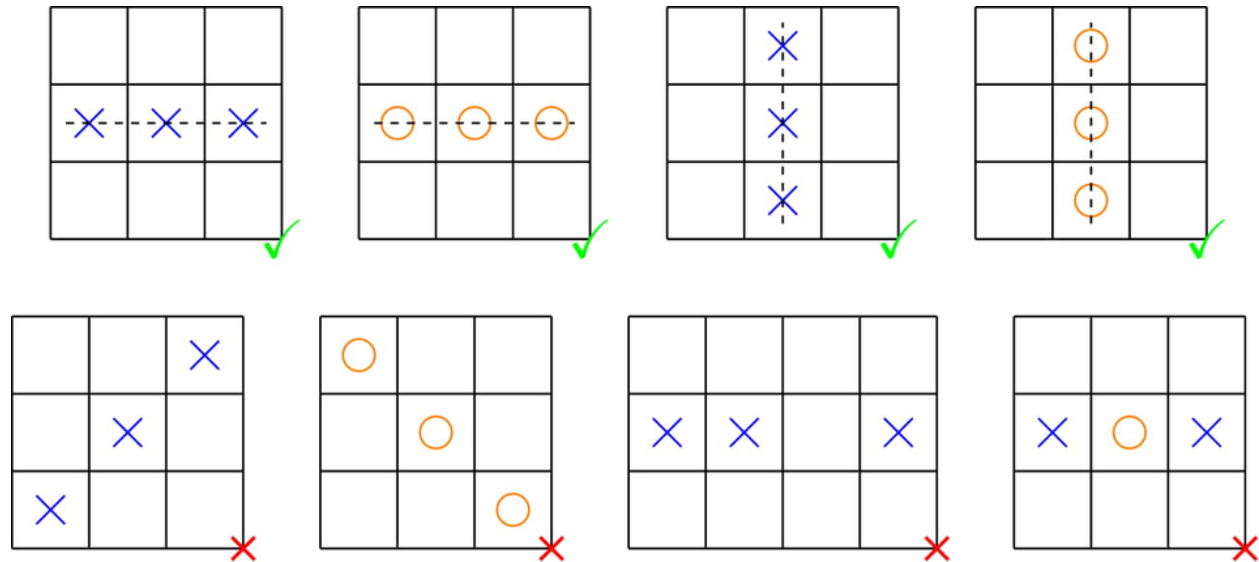


Figure 1 Minh họa trò chơi Tic tac toe

Minh họa trò chơi Tic tac toe

Nhiệm vụ của bạn là những ô đã đánh, thay đổi X thành O, hoặc O thành X sao cho hai bên đều hòa. Gọi K là số ô được điền. Số phép thay đổi không được vượt quá $\lfloor k/3 \rfloor$ (làm tròn xuống).

1.1 Đầu vào:

Dòng đầu chứa 1 số nguyên t ($1 \leq t \leq 100$) – số testcase.

Dòng đầu của mỗi testcase chứa n ($1 \leq n \leq 1000$) kích cỡ của ma trận vuông.

Theo sau là n dòng mỗi dòng chứa n ký tự ‘.’ hoặc ‘X’ hoặc ‘O’.

1.2 Đầu ra:

Với mỗi testcase, in lại ma trận sau khi thay đổi.

Có thể có nhiều giải pháp khác nhau dẫn đến kết quả hòa, chỉ cần in một trường hợp trong số đó.

1.3 Miền ràng buộc:

Bộ nhớ cho phép: 256MB

Thời gian chạy: 1s

Ví dụ:

Đầu vào	Đầu ra
1 3 .X. XXX .X.	.X. XOX .X.

Đầu vào	Đầu ra
3 3 .0. 000 .0. 6 XXX000 XXX000 XX..00 00..XX 000XX 000XX 5 .000. 0XXX0 0XXX0 0XXX0 .000.	.0. 0X0 .0. 0XX00X X0X0X0 XX..00 00..XX 0X0X0X X00XX0 .0X0. 00XX0 XX0XX 0XX00 .0X0.

Giải thích:

Ở ví dụ 1, ma trận 3x3, có 5 ô được điền, do đó $k = \lceil 5/3 \rceil = 2$, vậy nhiều nhất có thể thay đổi 1 ký tự, thay đổi X thành O ở hàng 2 cột 2 ta được một trường hợp thỏa mãn đề bài.

II. Nêu ý tưởng:

Nếu ma trận đơn thuần chỉ là toàn là X hoặc toàn là O ở các ô được điền, không mất tính tổng quát, giả sử bàn cờ nếu điền chỉ điền X, quan sát hai ma trận sau:

X	X	X	X	X	X			O	X	X	O	X	X	
X	X	X	X	X	X			X	X	O	X	X	O	
X	X	X	X	X	X			X	O	X	X	O	X	
X	X	X	X	X	X			O	X	X	O	X	X	
X	X	X	X	X	X			X	X	O	X	X	O	
X	X	X	X	X	X			X	O	X	X	O	X	

Gọi các đường chéo song song với đường chéo chính lần lượt là 0,1,2,3,4,5,6,7,8,9,10.

Nếu ta thay đường chéo 0, 3, 6, 9 thành cùng 1 loại là O thì ta được một bàn cờ hòa.

Lại gọi các đường chéo liên tiếp là 0,1,2 (phần dư của 3), ta có:

0	1	2	0	1	2
1	2	0	1	2	0
2	0	1	2	0	1
0	1	2	0	1	2
1	2	0	1	2	0
2	0	1	2	0	1

Như vậy, ta chỉ cần thay đổi các loại đường chéo 0 hoặc 1 hoặc 2 thành O thì sẽ thỏa mãn điều kiện đề bài, và ta cần số bước thay đổi nhỏ hơn K.

Chứng minh:

Gọi x_0 là số phần tử được điền ở các đường chéo loại 0.

Gọi x_1 là số phần tử được điền ở các đường chéo loại 1.

Gọi x_2 là số phần tử được điền ở các đường chéo loại 2.

Ta có: $x_0 + x_1 + x_2 = K$ (số phần tử được điền theo đề bài).

Như vậy, $\min(x_0, x_1, x_2) \leq K/3$. Vậy xóa $\min(x_0, x_1, x_2)$ sẽ thỏa mãn điều kiện đề bài.

Nếu ma trận có cả X và O, bài toán có thể sẽ thành ra như sau:

	O	O	O	
O	X	X	X	O
O	X	X	X	O
O	X	X	X	O
	O	O	O	

Ma trận đường chéo:

0	1	2	0	1
1	2	0	1	2
2	0	1	2	0
0	1	2	0	1
1	2	0	1	0

Như ở trên, ta cần mỗi đường chéo cùng một loại, là X hoặc là O. Cứ 3 đường chéo liên tiếp, ta chỉ cần nó chỉ ở hai loại trong 0,1,2 thì sẽ thỏa mãn điều kiện đề bài.

Chứng minh rằng buộc $k/3$:

Với mỗi đường chéo loại 0,1,2 có thể có phần tử $x_0, O_0; x_1, o_1; x_2, o_2$.

	0	1	2
X	X ₀	X ₁	X ₂
O	O ₀	O ₁	O ₂

Ta có: $X_0 + O_0 + X_1 + O_1 + X_2 + O_2 = K$

Do ta chọn 2 loại đường chéo để thay đổi, gọi A_{ij} là số phép chúng ta thay đổi X thành O loại đường chéo i và O thành X loại đường chéo j, thì $A_{ij} = X_i + O_j$.

Ta có: $A_{01} + A_{02} + A_{10} + A_{12} + A_{20} + A_{21} = 2K$

Vậy: $\min(A_{01}, A_{02}, A_{10}, A_{12}, A_{20}, A_{21}) \leq 2K/6 = K/3$. Thỏa mãn điều kiện đề bài.

Cuối cùng, ta có thuật toán như sau:

Gọi $\text{boardsX}[3][n][n]$ là một vector 3 chiều với $\text{boardsX}[0]$, $\text{boardsX}[1]$, $\text{boardsX}[2]$ là các phương án lấy đường chéo loại 0, loại 1 và loại 2 và thay nó bằng O (1).

Gọi $\text{boardsO}[3][n][n]$ là một vector 3 chiều với $\text{boardsO}[0]$, $\text{boardsO}[1]$, $\text{boardsO}[2]$ là các phương án lấy đường chéo loại 0, loại 1 và loại 2 và thay nó bằng X (2).

Gọi $\text{changesX}[3]$ là số phần tử cần thay đổi cho mỗi phương án $\text{boardsX}[0]$, $\text{boardsX}[1]$, $\text{boardsX}[2]$, và $\text{changesO}[3]$ là số phần tử cần thay đổi cho mỗi phương án $\text{boardsO}[0]$, $\text{boardsO}[1]$, $\text{boardsO}[2]$.

Gọi K là số phần tử tối đa cần thay đổi.

Duyệt trên ma trận với x hàng và y cột, nếu nó là '.' thì duyệt tiếp, nếu là X thì ta cập nhật phương án lấy đường chéo loại $(x+y)\%3$ và thay nó bằng O, tương tự với O. Tăng biến changesX , changeO .

Gọi minChanges là số phần tử nhỏ nhất khi thay đổi hai đường chéo.

Duyệt tổ hợp các sự kết hợp của (1) và (2), $\text{minChanges} = \min(\text{minChangege}, \text{changesX}[x] + \text{changesO}[y])$.

In trường hợp ma trận với điều kiện minChanges đã tìm được ở trên.

III. Minh họa thuật toán:

Giả sử với ma trận sau đây:

X	X	O
X	O	O
X	X	O

BoardsX[0], boardsX[1], boardsX[2] sau khi cập nhật:

BoardsX[0]	BoardsX[1]	BoardsX[2]																											
<table> <tr> <td><u>O</u></td><td>X</td><td>O</td></tr> <tr> <td>X</td><td>O</td><td>O</td></tr> <tr> <td>X</td><td><u>O</u></td><td>O</td></tr> </table>	<u>O</u>	X	O	X	O	O	X	<u>O</u>	O	<table> <tr> <td>X</td><td><u>O</u></td><td>O</td></tr> <tr> <td><u>O</u></td><td>O</td><td>O</td></tr> <tr> <td>X</td><td>X</td><td>O</td></tr> </table>	X	<u>O</u>	O	<u>O</u>	O	O	X	X	O	<table> <tr> <td>X</td><td>X</td><td>O</td></tr> <tr> <td>X</td><td>O</td><td>O</td></tr> <tr> <td><u>O</u></td><td>X</td><td>O</td></tr> </table>	X	X	O	X	O	O	<u>O</u>	X	O
<u>O</u>	X	O																											
X	O	O																											
X	<u>O</u>	O																											
X	<u>O</u>	O																											
<u>O</u>	O	O																											
X	X	O																											
X	X	O																											
X	O	O																											
<u>O</u>	X	O																											
changesX[0]=2	changesX[1]=2	ChangesX[2]=1																											

BoardsO[0], boardsO[1], boardsO[2] sau khi cập nhật:

BoardsO[0]	BoardsO[1]	BoardsO[2]																											
<table> <tr> <td>X</td><td>X</td><td>O</td></tr> <tr> <td>X</td><td>O</td><td><u>X</u></td></tr> <tr> <td>X</td><td>X</td><td>O</td></tr> </table>	X	X	O	X	O	<u>X</u>	X	X	O	<table> <tr> <td>X</td><td>X</td><td>O</td></tr> <tr> <td>X</td><td>O</td><td>O</td></tr> <tr> <td>X</td><td>X</td><td><u>X</u></td></tr> </table>	X	X	O	X	O	O	X	X	<u>X</u>	<table> <tr> <td>X</td><td>X</td><td><u>X</u></td></tr> <tr> <td>X</td><td><u>X</u></td><td>O</td></tr> <tr> <td>X</td><td>X</td><td>O</td></tr> </table>	X	X	<u>X</u>	X	<u>X</u>	O	X	X	O
X	X	O																											
X	O	<u>X</u>																											
X	X	O																											
X	X	O																											
X	O	O																											
X	X	<u>X</u>																											
X	X	<u>X</u>																											
X	<u>X</u>	O																											
X	X	O																											
ChangesO[0]=1	changesO[1]=1	changesO[2]=2																											

Như vậy: minChanges = 2.

Một đáp án có thể là ta đổi đường chéo loại 2 từ X thành O và đường chéo loại 0 từ O thành X. Đáp án:

X	X	O
X	O	<u>X</u>
<u>O</u>	X	O

IV. Mã nguồn:

```
1. def solve():
2.     n = int(input())
3.     board = [[] for _ in range(n)]
4.     boardX = [0,0,0]
5.     board0 = [0,0,0]
6.
7.     for i in range(3):
8.         boardX[i] = [[[for _ in range(n)] for _ in range(n)]
9.         board0[i] = [[[for _ in range(n)] for _ in range(n)]
10.
11.     changeX = [0,0,0]
12.     change0 = [0,0,0]
13.
14.     for x in range(n):
15.         board[x] = input()
16.         for y in range(n):
17.             for i in range(3):
18.                 boardX[i][x][y] = board[x][y]
19.                 board0[i][x][y] = board[x][y]
20.     k = 0
21.
22.     for x in range(n):
23.         for y in range(n):
24.             if board[x][y]=='.':
25.                 continue
26.             if board[x][y]=='X':
27.                 change0[(x+y)%3]+=1
28.                 board0[(x+y)%3][x][y]='0'
29.             else:
30.                 changeX[(x + y) % 3] += 1
31.                 boardX[(x + y) % 3][x][y] = 'X'
32.             k+=1
33.
34.     minChanges = 1e9
35.
36.     for i in range(3):
37.         for j in range(3):
38.             if i==j:
39.                 continue
40.             if (changeX[i]+change0[j]<=(int)(k/3)):
41.                 minChanges = min(minChanges,changeX[i]+change0[j])
42.
43.     for i in range(3):
44.         for j in range(3):
45.             if i==j:
46.                 continue
47.             if (changeX[i]+change0[j]==minChanges):
48.                 for x in range(n):
49.                     for y in range(n):
50.                         if board[x][y] == '.':
51.                             print('.',end='')
52.                         elif ((x+y)%3==i):
53.                             print(boardX[(x+y)%3][x][y],end='')
54.                         elif ((x+y)%3==j):
55.                             print(board0[(x+y)%3][x][y],end='')
56.                         else:
57.                             print(board[x][y],end='')
58.                 print()
```

```
59.         return
60.
61. if __name__ == '__main__':
62.     t = int(input())
63.     for _ in range(t):
64.         solve()
```

V. Tính toán độ phức tạp:

Trong trường hợp xấu nhất chương trình cần chạy $3 \times 3 \times N \times N = 9 \times N \times N$ để tìm được minChanges, vậy độ phức tạp là $O(N \times N)$.