

# HW 3：井井井井字棋

## Description

井字棋，又稱圈圈叉叉，是個十分有名的遊戲。然而，一般的井字棋只要雙方有使用最佳策略，最後一定會平手。因此，這裡介紹一個新的變種。由於是井字棋的加強版，我們稱它為「井井井井字棋」。

井井井井字棋的進行與井字棋有些相似，但是有兩個不同點：

- (1) 在5\*5的盤面上進行。（沒錯，「井井井井」的名稱就是這麼來的）
- (2) 每輪可以畫兩個（圈圈或叉叉）。

井井井井字棋第一輪一定是圈圈先走。因為每輪畫兩個，所以進行11輪之後，一共有22個格子被劃過了（留下三個空格），這時遊戲結束。而判斷勝負的方法如下：

- (1) 首先，將最後三個空格填成叉叉。
- (2) 對於五個橫列、五格直列和兩條對角線，如果該線上五個符號有至少四個相同，則代表該符號的玩家獲得一分。分數較高者獲勝。如果兩邊分數相同，則平手。

	O	O		

		X		
X	O	O		

O		X		
X	O	O		

O		X		
X	O	O		
				O
		O	O	X
			X	

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

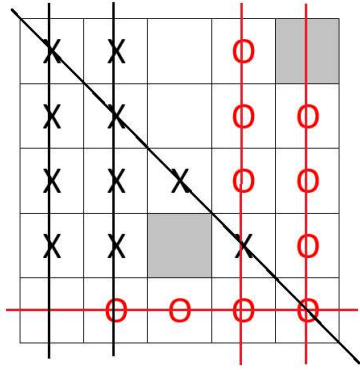
O	X	X		
X	O	O	X	
O	X	O	O	O
O	X	O	O	X
O	X	O	X	X

O	X	X
---	---	---

[這裡](#)有個簡單的互動式程式，可以自行編譯之後玩玩看這個遊戲。

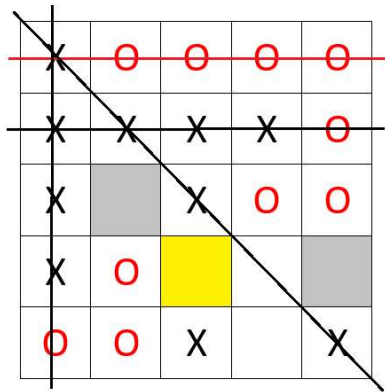
以下舉幾個「最佳策略」的例子：

(1)



這個例子中，下一輪是輪到圈圈。當前圈圈和叉叉各有三分，但是圈圈不可能再獲得任何分數，也就是他已經沒有勝算了。但是他卻可以下在以灰色標示的兩個格子，使遊戲平手。對於圈圈其他所有的策略，都會使得叉叉獲勝，因此這是圈圈的最佳策略，答案是平手。

(2)



這個例子中，下一輪是輪到圈圈。圈圈唯一拿到三分的方法是下在兩個黑色的位置，然而這樣會導致叉叉可以再多拿到一分（黃色格子）。所以這時圈圈必輸無疑，答案是叉叉獲勝。在這種情況也可以說，所有的下法都是最佳策略。

(3)

X	O	O	O	O
X	X	X	X	O
X		X	O	O
X	O	*		
O	O			*

這個例子中，下一輪是輪到叉叉。叉叉只要下在兩個灰色格子，就會變成(2)的局面，因此這是叉叉的最佳策略，答案是叉叉獲勝。  
從這個例子也可以發現，最佳策略不一定只有一種；在此例中，下在兩個星號的位置也會導致叉叉贏，因此這也是另一種最佳策略。

### Input / Output Format

請由stdin讀入、stdout輸出。

輸入的第一行有一個整數T，代表有T個盤面要判斷。  
接下來每五行代表一個盤面，每行都是一個長度為5的字串，只

由'O'、'X'、'.'構成。

對於每個盤面，如果最後會是圈圈贏，請輸出一行"O win"；如果最後會是叉叉贏，請輸出一行"X win"；如果最後會平手，請輸出一行"Draw"（皆不含引號）。

保證盤面一定是合法的，並且至少已經進行了4輪（也就是說，場上至少有四個圈圈和四個叉叉）。

### Sample Input

```
4
OXX..
XOOX.
OXOOO
OXOOX
OXOXX
XX.O.
XX.OO
XXXOO
XX.XO
.OOOO
XOOOO
XXXXO
X.XOO
XO...
OOX.X
XOOOO
XXXXO
X.XOO
XO...
OO...
```

### Sample Output

```
O win
Draw
X win
X win
```

### Subtasks

Subtask 1 (5 pt): 輸入已經進行了11輪。（也就是說，只要判斷到底是誰贏就好。）

Subtask 2 (15 pt): 輸入至少已經進行了10輪。

Subtask 3 (15 pt): 輸入至少已經進行了8輪。

Subtask 4 (15 pt): 輸入至少已經進行了7輪。

Subtask 5 (35 pt): 輸入至少已經進行了6輪。

Subtask 6 (10 pt): 輸入至少已經進行了5輪。

Subtask 7 (5 pt): 輸入至少已經進行了4輪。

### Suggestions & Hints

(1) 搜尋所有下一步有可能出現的盤面，並記錄它們的結果。假設當前是圈圈的回合，所有下一步的盤面中若有任何一個是圈圈必勝，那這個盤面也是圈圈必勝；如果都沒有圈圈必勝的話，找有沒有平手的盤面，若有，則是平手；如果還是沒有，那就是叉叉必勝了。以虛擬碼舉例：

```
WhoWin(board, round) {
    if (board is ended) return Evaluate(board);
    if (round == 'X') {
        result = "O win";
        for (nextboard in all possible next move) {
            nextresult = WhoWin(nextboard, 'O');
            if (nextresult == "X win") {
```

```
        result = "X win";
    } else if (nextresult == "Draw" && result == "O
win") {
        result = "Draw";
    }
}
return result;
} else {
    // ... likewise
}
}
```

(2) 搜尋過程中，有可能下的順序不同但導致了同一個盤面。可以用一個 **unordered\_map** 把所有已經算過的盤面記錄下來，這樣下次遇到這個盤面就不需要再算一次了。

(3) 這個盤面總共有25格，每一格有三種狀態（圈圈、叉叉或空白），因此每一格可以用兩個bit表示，也就是一個long long就足以表示整個盤面了。可以用這個方法把盤面的結果存在**unordered\_map**。

(4) 假設當前是圈圈的回合，如果搜到任何一個下一步的盤面是圈圈必勝，就可以直接確定這個盤面是圈圈必勝，而不必繼續搜尋下去了。再進一步想想看，有沒有可能在某些情況下，某個盤面到底是圈圈必勝還是平手其實完全不影響結果呢？（可以看成簡化版的[alpha-beta pruning](#)。）