

### ✓ 540. 孤單的機器人

Submit

Status

### Discussion / Solution

Ranklist

[Back to Problems List](#)

User's AC Ratio

Submission's AC Ratio

85.7% (12/14)

48.1% (13/27)

## Tags

Show solution-related tags

apcs

apcsc-loop



## Caido

主唱太拼命了

### Description

## 故事

很久很久以前，有一個機器人。它有著一個深愛它的主人 Stanley。在 Stanley 小的時候，常常對它下 WASD 的指令來操控這個機器人，只要輸入指令 W，機器人就會往上一格，A 是往左一格，S 是往下一格，D 是往右一格。雖然這機器人的功能陽春，但 Stanley 還是玩得不亦樂乎，與之度過了童年。

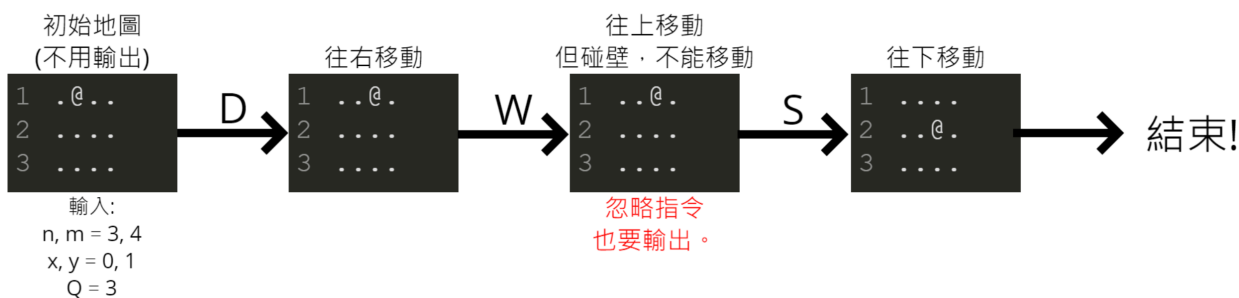
題目

輸入 Stanley 箱子的長  $m$  與寬  $n$ ，機器人的座標  $x, y$ ，以及接受到的指令數量  $Q$ ，請輸出每個指令接收後，箱子內部的樣子。

需要注意的是，要是執行指令後機器人會超出箱子的範圍，那麼機器人會自動忽略這個指令，但即便忽略了指令也要輸出現在箱子內部的樣子。

請用  表示機器人， 表示空地。

### 範例測資解釋



- 左邊的 1 2 3 是行號，不用輸出，對應的輸入輸出請見範例輸入輸出。

### Input Format

輸入有兩行：

第一行有五個數字，分別代表題目的  $n, m, x, y, Q$ 。

第二行有一個長度為  $Q$  的字串，每個字元依序代表著機器人的指令。

- $1 \leq n, m, Q$
- $0 \leq x < n$
- $0 \leq y < m$
- $1 \leq Qnm \leq 10^4$
- 字串的字元只會有 **WASD**，分別代表機器人的指令為上左下右。

### Output Format

對於每個指令，輸出其執行完後的箱字內部的狀態，請用 @ 表示機器人，. 表示空地。

### Sample Input 1

3 4 0 1 3  
DWS

### Sample Output 1

. . @ .  
 . . . .  
 . . . .  
 . . @ .  
 . . . .  
 . . . .  
 . . @ .  
 . . . .

Problem	Total Time (ms)	Max Memory (RSS, KiB) ?	Verdict ?	Score
540. 孤單的機器人	77	3456	Accepted	100

Subtask Results				^
Subtask no.	Testdata Range	Constraints	Score	
1	0	範例測資	0 / 0	
2	0~8	無額外限制	100 / 100	

Testdata Results							^
Testdata no.	Subtasks	Time (ms)	Memory (VSS, KiB) ?	Memory (RSS, KiB) ?	Verdict ?	Score	
0	① ②	2.7	6356	3456	Accepted	100	
1	②	4.1	6356	3456	Accepted	100	
2	②	4.6	6356	3456	Accepted	100	
3	②	7.3	6356	3456	Accepted	100	
4	②	8.1	6356	3456	Accepted	100	
5	②	6.5	6356	3456	Accepted	100	
6	②	18.8	6356	3456	Accepted	100	
7	②	4.7	6356	3456	Accepted	100	
8	②	20.4	6356	3456	Accepted	100	

```

char Q[num];

for (int i = 0; i < num; i++){

    cin >> Q[i];

}    //將每個指令存進陣列 Q。

if (Q[i] == 'W'){
    x -= 1;    //往上移動
    if (x < 0)        x = 0;

    else if (x > N - 1) x = N - 1;    //邊界檢查

if (x == 0 || x == N - 1){

    continue;    //如果到達邊界（最上或最下）則跳出迴圈印地圖

}

y -= 1;

if (y < 0) y = 0;

else if (y > M - 1) y = M - 1;    //A 往左移動（y -= 1）

```

$x += 1;$

if ( $x < 0$ )  $x = 0;$

else if ( $x > N - 1$ )  $x = N - 1;$     //S 往下移動 ( $x += 1$ )

$y += 1;$

if ( $y < 0$ )  $y = 0;$

else if ( $y > M - 1$ )  $y = M - 1;$     //D 往右移動 ( $y += 1$ )