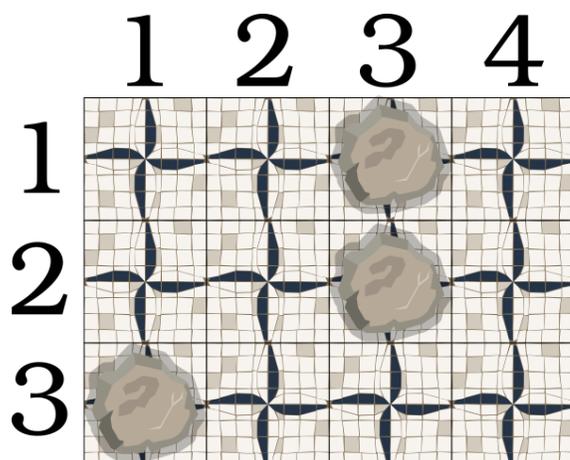
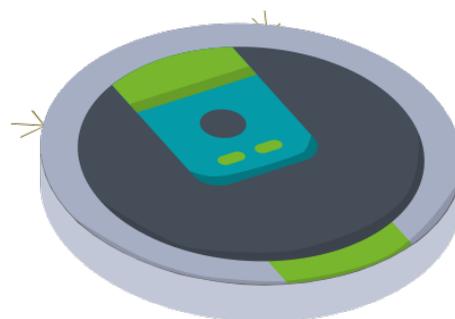


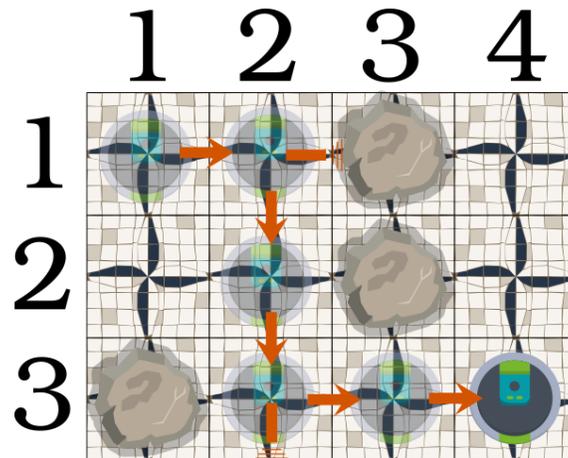
Problema A - Aspirar a Sala

Se é a primeira vez que participas, consulta a página de instruções para informações detalhadas sobre o formato deste problema.

A nova versão do ONI, o mais recente robô aspirador, está na fase final de desenvolvimento. Este robô representa uma sala como uma grelha de N linhas por M colunas, onde cada célula da grelha pode estar ocupada (representando um obstáculo) ou desimpedida, e à volta da sala existem paredes. É garantido que o canto superior esquerdo, ou seja, a posição $(1, 1)$, está sempre desimpedido. A imagem seguinte ilustra um exemplo para $N = 3$ e $M = 4$, onde as rochas representam células ocupadas e as restantes células estão desimpedidas.



O ONI é inicialmente colocado no canto superior esquerdo e pode movimentar-se pela grelha em 4 sentidos: para a direita, para a esquerda, para baixo ou para cima, e só se consegue movimentar para uma célula caso esteja desimpedida. Graças à melhor tecnologia de inteligência artificial portuguesa, o ONI consegue determinar um trajeto a seguir para limpar a sala eficientemente. Este trajeto consiste em K instruções que indicam a um robô ONI que se deve movimentar num dos 4 sentidos: direita, esquerda, baixo ou cima. Se o robô receber uma instrução que resulte num movimento para fora da grelha ou para uma célula ocupada, então o robô ignora essa instrução. Para o exemplo acima, suponhamos que um robô ONI começa no canto superior esquerdo e recebe as seguintes $K = 7$ instruções: direita, direita, baixo, baixo, baixo, direita, direita. A imagem seguinte ilustra o comportamento do robô para estas instruções:



O comportamento do robô é detalhado no seguinte:

- O robô começa por receber uma instrução de 'direita' e movimenta-se para a direita, passando a estar na posição (1, 2).
- De seguida, recebe outra instrução de 'direita', mas esta é ignorada visto que a célula (1, 3) está ocupada.
- Seguem-se duas instruções de 'baixo', que resultam no robô ocupar a posição (2, 2) e depois (3, 2).
- Segue-se mais uma instrução de 'baixo' que é ignorada visto que resultaria no robô passar a estar fora da grelha.
- Finalmente, as duas últimas instruções de 'direita' resultam na posição final, (4, 3).

Parte I

É dada a configuração de uma sala de N por M onde o canto superior esquerdo está desimpedido, assim como T conjuntos de instruções de movimentos, sendo que o i -ésimo conjunto tem K_i instruções. Assumindo que um robô ONI começa no canto superior esquerdo da sala, para cada conjunto de instruções determina as coordenadas da posição final do robô após completar todas as instruções. (ver a secção de Formato de Input para mais informação sobre como estes valores serão fornecidos)

Exemplo

Usando o exemplo anterior com $N = 3$, $M = 4$ e $K = 7$, a posição final do robô tem coordenadas (3, 4).

Restrições

São garantidos os seguintes limites em todos os casos de teste desta parte que irão ser colocados ao programa:

- $1 \leq N, M \leq 100$ Tamanho da sala
- $1 \leq T \leq 10$ Número de conjuntos de instruções
- $1 \leq K_i \leq 100$ Número de instruções

Os casos de teste desta parte do problema estão organizados em dois grupos:

Grupo	Número de Pontos	Restrições adicionais
1	15	$M = 1$
2	25	Sem restrições adicionais

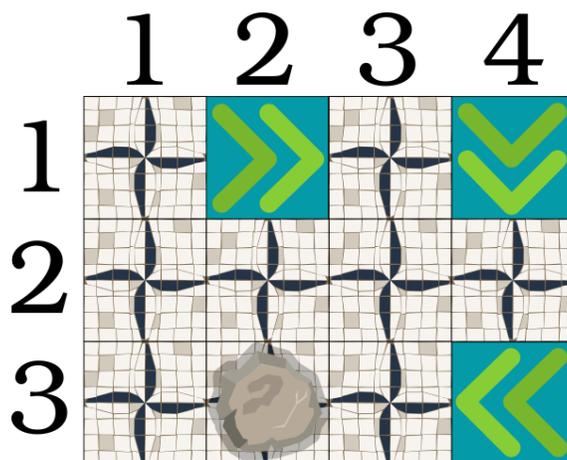
Parte II

Há também um conjunto de células especiais que contêm um sentido forçado. Quando um robô ONI pisa uma destas células entra em modo “automático”, ou seja, continua a seguir o sentido dado pela célula especial até: o próximo movimento resulte no robô passar a estar fora da grelha ou numa célula ocupada, neste caso para de andar e volta a ter controlo sobre o seu movimento, ou seja sai do modo “automático”; ou robô movimenta-se para uma célula especial, neste caso passa a seguir o sentido da nova célula especial que acaba de pisar. Nota que é possível que o robô entre em ciclo, ou seja, que o robô não saia mais do modo automático após pisar uma célula especial.

É dada a configuração de uma sala de N por M onde o canto superior esquerdo está desimpedido, assim como T conjuntos de instruções de movimentos, sendo que o i -ésimo conjunto tem K_i instruções. Assumindo que um robô ONI começa no canto superior esquerdo da sala, para cada conjunto de instruções determina se o robô entra em ciclo ou, caso contrário, as coordenadas da posição final do robô após completar todas as instruções.

Exemplo

Considera a seguinte sala com $N = 3$, $M = 4$.



Se tivermos as seguintes $K = 3$ instruções: direita, direita, direita, então o robô segue o seguinte percurso, terminando na célula (3,3).

$1 \leq N, M \leq 100$ Tamanho da sala
 $1 \leq T \leq 10$ Número de conjuntos de instruções
 $1 \leq K_i \leq 100$ Número de instruções

Os casos de teste desta parte do problema estão organizados em dois grupos:

Grupo	Número de Pontos	Restrições adicionais
3	25	$M = 1$
4	35	Sem restrições adicionais

Sumário de subtarefas

Os casos de teste do problema estão organizados em quatro grupos com restrições adicionais diferentes:

Grupo	Número de Pontos	Parte	Restrições adicionais
1	15	Parte I	$M = 1$
2	25	Parte I	Sem restrições adicionais
3	25	Parte II	$M = 1$
4	35	Parte II	Sem restrições adicionais

Formato de Input

A primeira linha contém um inteiro P , que representa a parte que o caso de teste representa. Se for 1, então o caso de teste refere-se à Parte I, se for 2 então refere-se à Parte II.

Segue-se uma linha com dois inteiros separados por espaços, primeiro N , representando o número de linhas da grelha, seguido de M o número de colunas.

Seguem-se N linhas, cada uma com M caracteres, representando a grelha. Um '.' é usado para representar uma célula desimpedida e um '#' é usado para representar uma célula com um obstáculo. Para a Parte II também podemos ter os caracteres 'D' (direita), 'E' (esquerda), 'B' (baixo), 'C' (cima), representando células especiais com o sentido indicado.

Segue-se uma linha, contendo um inteiro T , representando o número de conjuntos de instruções.

Seguem-se T conjuntos de linhas, cada uma seguindo o formato seguinte:

Uma linha com um inteiro K , o número de instruções.

Uma linha com uma sequência de K caracteres sem espaços entre eles, representando instruções. Os caracteres podem ser 'D' (direita), 'E' (esquerda), 'B' (baixo), 'C' (cima), representando uma instrução no sentido indicado.

É garantido que o canto superior esquerdo de todas as grelhas será sempre uma célula desimpedida.

Formato de Output

Parte I

O output deve conter T linhas com dois inteiros separados por espaços cada, representando a linha e a coluna da célula onde o robô termina de executar todas as instruções.

Parte II

O output deve conter T linhas contendo ou a palavra 'ciclo', se o robô entrar em ciclo, ou dois inteiros separados por espaços cada, representando a linha e a coluna da célula onde o robô termina de executar todas as instruções.

Nota: deve existir exatamente um único espaço entre cada inteiro e não deve haver nenhum espaço no final da linha (ou seja, após o último inteiro deve aparecer apenas uma mudança de linha). Se este formato não for respeitado o resultado de uma submissão será **Presentation Error** (consulta as instruções para mais informações).

Input do Exemplo 1

```
1
3 4
..#.
..#.
#...
1
7
DDBBDD
```

Output do Exemplo 1

```
3 4
```

Explicação do Exemplo 1

Este exemplo corresponde ao exemplo da Parte I mencionado no enunciado.

Input do Exemplo 2

```
2
3 4
.D.B
....
.#.E
1
3
DDD
```

Output do Exemplo 2

```
3 3
```

Explicação do Exemplo 2

Este exemplo corresponde ao exemplo da Parte II mencionado no enunciado.

Input do Exemplo 3

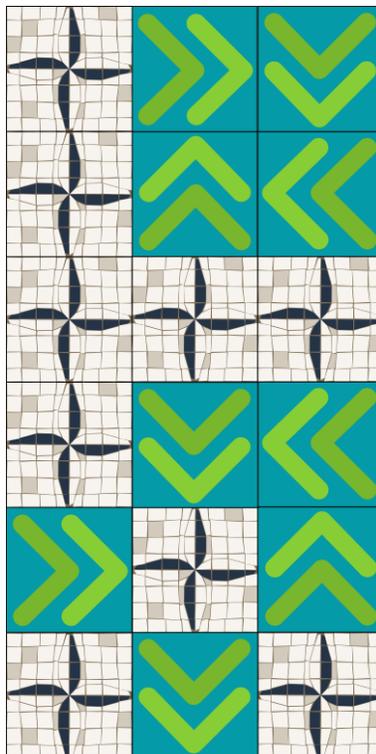
```
2
6 3
.DB
.CE
...
.BE
D.C
.B.
2
4
DEBC
5
BBBBD
```

Output do Exemplo 3

```
ciclo
6 3
```

Explicação do Exemplo 3

A grelha do exemplo anterior corresponde ao seguinte:



Organização



Patrocinadores



Apoios

