**02.Survivor**

*Welcome to the new elimination Wednesday in Survivor. Your task today is to find all the tokens from the surrounding beach. But better do it before the other contestants can, you don’t want to risk getting eliminated.*

Write a program, that collects tokens from the beach. First you will be given the number of **rows** of the beach – an **integer n**. On the next **n** lines, you will receive the **available tokens** to **collect** for **each row**, **separated by a single space** in the format:  
**"{token1} {token2} … {tokenn}"**  
The **positions** (cells) **without tokens** in them are considered **empty** and they will be **marked** with a **dash ('-').**

After that you will start receiving **commands**. There are **three** possibilities:

Добре дошли в новото елиминацията в сряда в Сървайвър. Вашата задача днес е да намерите всички токени от околния плаж. Но по-добре да го направя преди другите състезатели да могат, не искате да рискувате да бъде елиминиран.

Напиши програма, която събира жетони от плажа. Първо ще ви бъде даден брой редове на плажа – цяло число n. На следващите n редове ще получите наличните символи за събиране за всеки ред, разделени с едно място във формата: "{token1} {token2} ... {tokenn}" Позициите (клетките) без маркери в тях се считат за празни и ще бъдат маркирани с тире ("-"). След това ще започнете да получавате команди. Има три възможности:

* **"Find {row} {col}"**:
  + You have to go to the given place **if it is valid** and collect the token, **if** there is one.
  + When you **collect** it, you have to mark the place as an **empty**, using **dash** symbol.
  + "Намери {row} {col}": Трябва да отидете на дадено място, ако е валидно и да съберете маркера, ако има такъв.
  + Когато го събирате, трябва да маркирате мястото като празно, като използвате тире символ.
* **"Opponent {row} {col} {direction}"**:
  + One of your opponents **is searching for a token** at the given coordinates **if they exist**.
  + After **going** at the given **coordinates** (**if they exist**) and **collecting** the **token** (**if there is such**), **the opponent** is beginning a **movement** in the **given direction** by **3 steps**. He **collects** all the **tokens** that are **placed on his way**.
  + If **opponent's movement** is going to **step outside of the field**, he is **stepping only** on **the possible indexes**.
  + When he **finds** tokens, he **marks the cells as** **empty**.
  + There are four possible directions, in which he can go: **"up", "down", "left", "right".**
  + Опонент {row} {col} {direction}": Един от опонентите ви търси знак на дадените координати, ако те съществуват.
  + След като отива на дадените координати (ако те съществуват) и събиране на токен (ако има такъв), противникът започва движение в дадената посока с 3 стъпки. Той събира всички символи, които са поставени по пътя му.
  + Ако противникът ще излезе извън полето, той се стъпва само на възможните индекси.
  + Когато намери жетони, той маркира килиите като празни.
  + Има четири възможни посоки, в които той може да отиде: "нагоре", "надолу", "ляво", "дясно".
* **"Gong"** - the gong rings and the challenge is over.

In the end, print on the console the **last condition of the beach**. The cells, containing a token or not, should be **separated by single space**. After that print the **count** of the tokens you've collected:  
"**Collected tokens: {countOfCollected}**"

Last step is to print the number of **found by your opponent tokens** in the format:  
"**Opponent's tokens: {countOfOpponentsTokens}**"

**"Гонг" - пръстените и предизвикателството свърши.**

**В крайна сметка, отпечатайте на конзолата последното състояние на плажа. Клетките, съдържащи жетон или не, трябва да бъдат разделени с едно място. След това отпечатайте броя на жетоните, които сте събрали: "Събрани жетони: {countOfCollected}" Последната стъпка е да отпечатате броя намерени от вашите жетони на противника във формат: "Жетоните на противника: {countOfopponentsTokens}"**

## Input

* On the first line, you will receive the number of beach's **rows** - **integer n**
* On the next **n** lines, for each row, the situation of the **seashells** at the beach in the described **format** above
* Next, until you receive **"Gong"**, you will get the **commands** in the specified format.
* На първия ред ще получите броя на редовете на плажа - цяло число n
* На следващия n линии, за всеки ред, ситуацията на морските костенурки на плажа в описания по-горе формат, докато получите "Гонг", ще получите командите в посочения формат.

## Output

* Print the **resulting** beach - each cell separated **by single space**
* On the next output line - print information for **tokens** you've **collected** in the **described** format
* On the last line - print the **number** of tokens found by the opponent

## Constraints

* The number of **rows** will be **positive** **integer** between **[1, 10]**
* Not all **rows** will have **the same length**
* The **tokens** be marked with **'T'**
* Move commands will be: "**up**", "**down**", "**left**", "**right**"
* Броят на редовете ще бъде положително цяло число между [1, 10]
* Не всички редове ще имат една и съща дължина
* Символите се маркират с команди 'T'
* Move ще бъдат: "нагоре", "надолу", "ляво", "надясно"

## Examples

|  |  |  |
| --- | --- | --- |
| ****Input**** | ****Output**** | ****Comment**** |
| **6**  **T T - T T - T**  **- T - -**  **T - T - T T - -**  **- T - T - T**  **T T**  **T T T - T**  **Find 2 2**  **Find 4 1**  **Opponent 3 1 up**  **Find 4 3**  **Find 5 0**  **Find 4 0**  **Opponent 2 0 down**  **Gong** | **T - - T T - T**  **- - - -**  **- - - - T T - -**  **- - - T - T**  **- -**  **- T T - T**  **Collected tokens: 4**  **Opponent's tokens: 4** | First we receive two **"Find"** commands, we go to the given coordinates, collect the **'T'** and leave its cells **empty ('-')**. After that there is **"Opponent"** command – your opponent **goes** at coordinates **3 1**, first collects **'T'**, then takes **3 steps up** - the first cell is **empty**, so he continues up, on the **second** step he steals **'T'** and on the **third** - **'T'** and sets their cells as **empty**. The **"Find"** command is next, but we **don't do** anything, because the coordinates are **invalid**. We execute the last commands in the same way. In the end we print the beach. We've collected **4** tokens. Your opponent managed to **find** **4** tokens.  Първо получаваме две команди "Намери", отиваме на дадените координати, събираме "T" и оставяме клетките празни ("-"). След това има команда "Опонент" – противникът ви отива на координати 3 1, първо събира 'T', след това се вдигат 3 стъпки нагоре - първата клетка е празна, така че продължава нагоре, на втората стъпка той краде 'T' и на третото - 'Т' и определя клетките им като празни. Командата "Намери" е следваща, но ние не правим нищо, защото координатите са невалидни. Изпълняваме и последните команди по същия начин. В крайна сметка ние печат на плажа. Събрахме 4 жетона. Опонентът ти успя да намери 4 жетона |
| 4  - T T  T  T - - -  T  Find 9 0  Find 1 4  Opponent 0 2 right  Opponent 5 5 up  Gong | **- T -**  **T**  **T - - -**  **T**  **Collected tokens: 0**  **Opponent's tokens: 1** |  |