## Miner

You are going to create a game called "Miner".

First, you will receive the **size** of a **square field** in which the miner should move.

On the second line, you will receive the **commands** for the miner's movement, separated by a single space. The possible commands are **"left"**, **"right"**, **"up"** and **"down"**.

In the end, you will receive **each row of the field** on a separate line. The possible characters that may appear on the screen are:

* **\*** - a regular position on the field
* **e** - the end of the route
* **c** - coal
* **s** - miner

The **miner** **starts moving** from the position **"s"**. He should perform the given commands **successively**, moving with **only one position** in the given direction. If the miner has reached the **edge of the field** and the following command indicates that he has to get out of the area, he must **remain in his current position** and **ignore** the command.

When the miner finds **coal**, he **collects it** and **replaces i**t with **"\*"**. Keep track of the collected coal. In the end, you should print whether the miner has **succeeded** **in collecting the coal** or not and his **final position**:

* If the miner has **collected all coal in the field**, the program stops, and you should **print** the message: "**You collected all coal! ({row\_index}, {col\_index})"**.
* If the miner **steps at** **"e"**, the game is **over** (the program stops), and you should **print** the message: **"Game over! ({row\_index}, {col\_index})"**.
* If there are **no more commands** and **none** of the above cases had **happened**, you should **print** the message: **"{number\_of\_remaining\_coal} pieces of coal left. ({row\_index}, {col\_index})"**.

### Input

* **Field size** - an integer number
* **Commands to move** the miner - a sequence of directions, separated by a single whitespace (**" "**)
* **The field: some of the following characters ("\*", "e", "c ", "s"),** separated by a single whitespace (**" "**)

### Output

* There are three types of output as mentioned above.

### Constraints

* The **field size** will be a 32-bit integer in the range **[0 … 2 147 483 647]**
* The field will always have only one **"s"**

### Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| 5  up right right up right  \* \* \* c \*  \* \* \* e \*  \* \* c \* \*  s \* \* c \*  \* \* c \* \* | Game over! (1, 3) |
| 4  up right right right down  \* \* \* e  \* \* c \*  \* s \* c  \* \* \* \* | You collected all coals! (2, 3) |
| 6  left left down right up left left down down down  \* \* \* \* \* \*  e \* \* \* c \*  \* \* c s \* \*  \* \* \* \* \* \*  c \* \* \* c \*  \* \* c \* \* \* | 3 coals left. (5, 0) |