# Problem 2 - Snake

One of the candidates-intern must solve the task - "Snake". He must write a Java code that calculates wheater the snake wins or loses the game.

Here are the rules of the game:

The game starts with a **snake length of 1**.

We get as input **the size** of the **screen** in which our snake moves. The screen is **always a square**. After that we receive the commands wich represent the directions in wich the snake should move.The snake **starts** from **s**-position The commands will be: **left/right/up/down.** If the snake reaches the side edge of the screen (left or right), it goes to the **opposite side of the same row**. If the snake reaches the top/bottom edge of the screen it goes on the **opposite site of the same column** . The possible characters that may appear on the screeen are:

* **\***– that is a regular square; it does nothing
* **e** – represents an enemy.
* **f** – this is the food
* **s** - the place where the **game starts**

Each time when you eat a piece of food your **length increases by one**. Keep thrack of the length, because **in case you win you have to print it**. If you **step on an enemy the game is over (the snake stops moving)** and you have to print the output as shown in the output section. After executing all of the commands there are 3 possible outcomes:

* you have eaten all the food and you win
* you get killed by an enemy
* there is still some food to be eaten

Print the corresponding output depending on the case.

## Input

* **Length** of the screen side – an integer number.
* **Commands to move** the snake – an array of strings separated by **“, ”**.

## Output

* There are three types of output:
  + If all of the food is eaten print the following output: “**You win! Final snake length is {length}**”
  + If there are no left commands and there is still some food to be eaten: “**You lose! There is still {left food} food to be eaten.**”
  + If you step on the enemy the game is over and you print “**You lose! Killed by an enemy!**”

## Constraints

* The **input numbers** will be a 32-bit integer in the range [0 … 2 147 483 647].
* Allowed working time for your program: 0.1 seconds.
* Allowed memory: 16 MB.

## Examples

|  |  |  |
| --- | --- | --- |
| **Input** | **Output** | **Comments** |
| 5  up, right, right, right, up  \* \* e \* \*  \* \* \* f \*  \* f \* \* \*  s \* \* \* \*  \* \* e \* \* | You win! Final snake length is 3 | After executing all of the commands the snake has eaten all of the food and it it still alive. |
| 4  right, right, right, right, right, down, right  \* s \* \*  \* \* e \*  \* f \* f  \* \* \* f | You lose! Killed by an enemy! | The snake moves 2 times to the right, then it goes off screen and appears on left side. Then it makes 2 more moves to the right, it goes down, steps on an enemy and the game is over. |
| 6  down, left, left, down, right, right, right, right, right  \* \* \* \* s \*  \* e f \* \* \*  f \* \* \* \* \*  \* \* \* f e \*  \* e \* \* \* \*  \* \* \* \* \* \* | You lose! There is still 1 food to be eaten. | The snake survived but there is still 1 piece of food that the snake could not eat. |