## Problem 2 – Ani is drunk

Ani is drunk as boot. She is at the disco and she needs to throw up like immediately. However, she is a civilized young lady so she wants to do it in the toilets. Unfortunately, at the disco all the toilet cabins are always full. So, she has to spend quite some time to find a free cabin. She is also hearing a voice that tells her how many steps she needs to go to the next cabin to check. There is a problem, though. The voice is telling her to go to the right only, and the number of steps is more than the number of cabins. The idea is that if she is at the last cabin and has to go a few more steps, she starts over at the first cabin.

She is so drunk that she can’t math. Help her with the calculations.

The first cabin is always at number **zero**, the last is **numberOfCabins - 1**

For example, we have 5 cabins; she is currently at cabin #2, and needs to go 4 steps to the right. She passes by cabin #3 and #4, and reaches the end. She needs to start over at **#0** and stops at cabin **#1**. She didn’t have to go so many steps; she could just move from cabin **#2** to cabin **#1**. Your task is to calculate the next cabin and tell her which direction to go and how many steps to take.

### Input

At the first line, you will receive an integer **N –** the number of cabins.  
At the next **K** lines, you will receive an integer **S** – the number of steps she need to take.  
At the last line, you will receive the command “Found a free one”, which denotes end of the input.

### Output

For each **K** line, print the actual number of steps that she needs to take, in one of the following formats:

* “Go {0} steps to the right, Ani.”
* “Go {0} steps to the left, Ani.”
* “Stay there, Ani.”

At the last line, you must print to total number of steps done by Ani, in format:

* “Moved a total of {0} steps.”

### Constraints

* The number **N** is an integer in the range [0… 4,000,000,000] inclusive.
* The number **S** is an integer in the range [0… 4,000,000,000] inclusive.
* There will be no more than 20,000 lines of **K**
* Allowed work time for your program: 0.1 seconds.
* Allowed memory: 16 MB.

### Examples

|  |  |  |
| --- | --- | --- |
| **Input** | **Output** | **Comments** |
| 5  8  3  4  1  8  Found a free one! | Go 3 steps to the right, Ani.  Go 2 steps to the left, Ani.  Go 1 steps to the left, Ani.  Go 1 steps to the left, Ani.  Go 3 steps to the right, Ani.  Moved a total of 10 steps. |  |

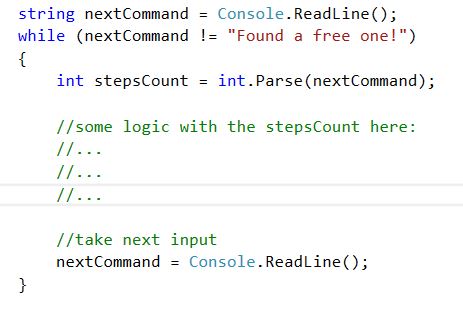
### Instructions

First, let’s take a look at the input. At the first line, we will receive the number **N** , representing the number of cabins. We need to read that number and store it in a variable:

C:\Users\Edu\Desktop\New folder\intparse.JPG

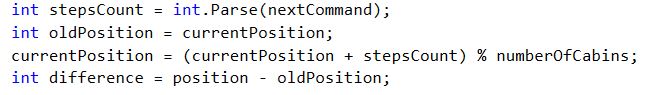
Next, we will receive an unknown number of lines **K** that will end with the message **“Found a free one!”**  To handle this, we can use a **while loop** with the appropriate condition. This reads as:

***While*** the input is different from “Found a free one!” then it must be the next number of steps. ***When*** we receive the command “Found a free one!” - exit the loop and continue with the next part of the program

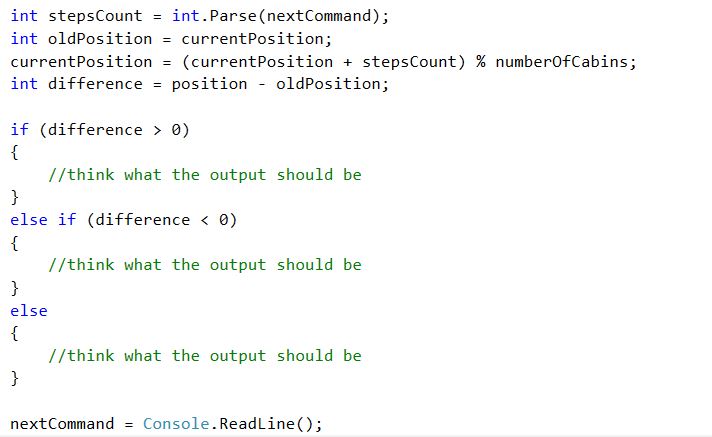


To keep track of the current position, we can initialize a variable outside the while loop and call it **currentPosition.** At the start, its value is 0. In the while loop, we need to add the next number of steps and also keep the **currentPosition** in range if its value is greater than the **numberOfCabins**. We can use the % operator for that purpose.C:\Users\Edu\Desktop\New folder\position.JPG

We also need to know how the **currrentPosition** has changed in relation with the old position. We can store the old position in an appropriate variable



We are almost ready with the task. Next thing we must do is send the instructions to Ani. We will use Console.WriteLine();



You must also keep track of the total number of actual steps that Ani has done. For that you should create another variable at the beginning of the program. Call it **totalLengthCovered,** for example. Its initial value is 0. Take note that the value in the **difference** variable could be negative, but the number of steps the Ani takes is always positive. You can use **Math.Abs(difference)** to avoid possible mistakes.

