## 01.Spellbook Unraveling

**Environment Specifics**

Please be aware that every JS environment may **behave differently** when executing code. Certain things that work in the browser are not supported in **Node.js**, which is the environment used by **Judge**.

The following actions are **NOT** supported:

* **.forEach()** with **NodeList** (returned by **querySelector()** and **querySelectorAll()**)
* **.forEach()** with **HTMLCollection** (returned by **getElementsByClassName()** and **element.children**)
* Using the **spread-operator** (**...**) to convert a **NodeList** into an array
* **append()** in Judge (use only **appendChild()**)
* **prepend()**
* **replaceWith()**
* **replaceAll()**
* **closest()**
* **replaceChildren()**
* Always turn the collection into a **JS array** (forEach, forOf, et.)

If you want to perform these operations, you may use **Array.from()** to first convert the collection into an array.

*You find yourself in a dimly lit chamber filled with ancient tomes and mystical artifacts. The air crackles with arcane energy as you embark on a journey to decipher the secrets hidden within the pages of a spellbook. Each incantation holds the potential to unleash powerful magic or unveil hidden truths.*

Write a program that performs a series of commands in order to decode that information. First, you will receive a **string** representing a mysterious spell waiting to be unraveled, and afterwards, until the command "**End**" is given, you will be receiving strings with commands **split** by a **exclamation mark**. The **commands** will be the following:

## **"RemoveEven"**

* + The encoded spell is changed to consist only of its characters at **even** **indices.** Then, the updated spell is **printed**.

## **"TakePart!{from index}!{to index}":**

* + Selects a portion of the spell specified by two numbers, unveiling a segment of its hidden power and then **prints** the spell.

## **"Reverse!{substring}":**

* + If the spell contains the given **substring**, **cut it out**, **reverse** it and **add** it at the **end** of the spell. Then, **print** the updated spell.
  + Otherwise, print "Error".
  + This operation should replace only the first occurrence of the given **substring** **if there are two or more occurrences**.

## Input

* You will be receiving strings until the "**End**" command is given.

## Output

* After the "End" command is received, **print**:
  + "The concealed spell is: {spell}"

## Constraints

* All given **commands** will be **valid**.
* **Aways** will recive encoded spell

## Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| (["asAsl2adkda2mdaczsa",  "RemoveEven",  "TakePart!1!9",  "Reverse!maz",  "End"]) | aAlakamaza  Alakamaz  Alakazam  The concealed spell is: Alakazam |
| **Input** | **Output** |
| (["hZwemtroiui5tfone1haGnanbvcaploL2u2a2n2i2m",  "TakePart!31!42",  "RemoveEven",  "Reverse!anim",  "Reverse!sad",  "End"]) | L2u2a2n2i2m  Luanim  Lumina  Error  The concealed spell is: Lumina |