# Disneyland Journey



*You are planning a trip to Disneyland in the next year. Are you able to save money for it?*

Create a program that checks if you can **save the money** for the Disneyland’s journey. You have a **certain number of months** for which you can collect the money.

**At the end of each month,** you save **25% of the cost of the journey.**

**At the beginning of** every **odd** month (**except** the **first**one) you **spent 16%** of the money collected so far for clothes and shoes.

Every **4th (fourth)** month **at the beginning of the month** your bossgives you а bonus. It is **25%** of the money collected so far.

If you save enough money for the journey, calculate how much money will be left for the souvenirs. Then print the following:

**"Bravo! You can go to Disneyland and you will have {money}lv. for souvenirs."**

If the saved money is less you should calculate how much money you need more. Then print the following:

**"Sorry. You need {money}lv. more."**

Both numbers should be **formatted** to the **2nd decimal place**.

## Input

* On the **1st line** you will receive how much the journey will cost – a **real number** in the range [500.0…10000.0]
* On the **2nd line** you will receive the **number of months** for which you have to collect money – an **integer number** in the range [1…12]

## Output

* Print the output in the **format** **described** **above**.

## Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| 1000 4 | Bravo! You can go to Disneyland and you will have 87.50lv. for souvenirs. |
| **Comments** | |
| You need 1000 leva for the journey and you have 4 months to collect them. Every month you can save 1000 \* 25% => 250 lv. So, to the end of the **1st** month you have **250 lv**. To the **end** of the **2nd** month - 250 + 250 -> **500** **lv**. To the **begging** of the **3th** month (**odd month**) you spent **80 lv.** (500 \* 16%) for clothes and shoes, then you save **250 lv**, so now you have **670 lv**. Last month is the **fourth month** and you save 670 + 167.5 (670 \* 25%) + 25 = **1087.5 lv**. You have 1087.5 – 1000 = **87.5 lv.**, so you can go to the trip. | |
|  | |
| 3265  3 | Sorry. You need 1077.45lv. more. |

# Archery Tournament

*Our hero Iskren is going to take part in an archery tournament. Your task is to help him calculate his points.*

On the **first line** you will receive integers separated by **"|"**, representing the targets in the archery field.  
On the next lines until you receive "**Game over**" command, you will receive commands by the Judge of the tournament:

* **"Shoot Left@{start index}@{length}":**
* Iskren starts traversing the archery field to the **left** from **{start index}** with given **{length}**.
* If he goes **out of the field**, he will continue from the **end of the field**.
* **"Shoot Right@{start index}@{length}":**
* Iskren starts traversing the archery field to the **right** from **{start index}** with given **{length}**.
* If he goes **out of the field**, he will continue from the **start of the field.**
* **"Reverse":**

**Reverse** the archery field.

* **"Game Over"**

**Print** the archery **field and collected points**.

When **he arrives at the target**, he will shoot at it and **increase his points by 5** and **decrease the target by 5 points**, if the **target points are less than 5**, he takes **all of them and decreases it to 0**. If the **start index is out of range** of the field Iskren will have to **ignore the command**.

## Input

* On the **first line**, you will receive integers **separated** by **"|"** representing the **targets in the archery field**.
* On the **next lines**, until the "**Game over**" command you will receive **commands** in the **format described** **above**.

## Output

* Print the field in following format: **"{target} - {target} - {target} ….. - {target}"**.

**"Iskren finished the archery tournament with {points}!"**

## Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| 10|10|10|10|10 Shoot Left@0@2  Shoot Right@4@5  Shoot Right@6@5  Reverse  Game over | 5 - 5 - 10 - 10 - 10  Iskren finished the archery tournament with 10 points! |
| **Comments** | |
| First, Iskren receives the **"Shoot Left@0@2"** command, so he starts traversing the archery field from **index 0** with **length 2** and he stops at **index 3**. He shoots and the archery filed looks like this: 10 - 10 - 10 - 5 – 10 ,  and his points are 5.  Then he receives **"Shoot Right@4@5"** -> **"10 - 10 - 10 - 5 – 5"**  **"Shoot Right@6@5"** -> Index 6 is **out of range** of the field , so Iskren ignores the command.  **“Reverse”**-> After that command the field looks like this: 5 - 5 - 10 - 10 - 10 | |
|  | |
| 20|30|40|50|60 Shoot Left@0@12  Shoot Right@4@15  Shoot Left@6@5  Reverse  Game over | 55 - 45 - 40 - 30 - 20  Iskren finished the archery tournament with 10 points! |

# School Library

*As a young developer Iskren is a part from software development school team. His task is to do online book library, but he needs your help for that.*

On the first line you will receive a String, representing shelf with books in library. Every book is separated with "**&**".

On the next lines until you receive "**Done**" command, you will receive following commands:

* **Add Book | {book name}**
* **Add** a book at **first place** in the shelf.
* If the book **already is present** on the shelf, **ignore the command**.
* **Take Book | {book name}**
* **Remove** the book with the given name only **if the book is on the shelf**, otherwise **ignore this command**.
* **Swap Books | {book1} | {book2}**
* If both books **are on the shelf**, swap their places.
* **Insert Book | {book name}**
* **Add** a book at the **end of the book collection**.
* **Check Book | {index}**
* **Print** the name of the book on the given index the book.
* If the index is **invalid**, **ignore the command**.

## Input

* **On the 1st line**, you will receive a string, representing book names, separated with "**&**".
* On the **next lines**, until you receive "**Done**", you will receive **commands** in the **format described** above.

## Output

* Print the **collection** of books joined by **", ".**

**"{firstBook}, {secondBook}, …{lastBook}"**

## Constraints

* You won't receive duplicate book names in the initial list of books.

## Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| Don Quixote&The Great Gatsby&Moby Dick&Hamlet  Add Book | The Odyssey  Take Book | Don Quixote  Insert Book | Alice's Adventures in Wonderland  Check Book | 3  Done | Hamlet  The Odyssey, The Great Gatsby, Moby Dick, Hamlet, Alice's Adventures in Wonderland |
| **Input** | **Output** |
| Anna Karenina&Heart of Darkness&Catch-22& The Stranger  Add Book | David Copperfield  Add Book | One Thousand and One Nights  Swap Books | One Thousand and One Nights | Catch-22  Take Book | David Copperfield  Insert Book | The Stories of Anton Chekhov  Check Book | 17  Done | Catch-22, Anna Karenina, Heart of Darkness, One Thousand and One Nights, The Stranger, The Stories of Anton Chekhov |