# 02. Array Modifier

You are given **an array with integers**. Write a program to **modify the elements** after **receive the commands** “**swap**”, “**multiply**” or “**decrease**”.

* “swap {index1} {index2}” take **two elements** and **swap their places**.
* “multiply {index1} {index2}” take **element at the 1st index** and **multiply** **it** **with element at 2nd index**. **Save the product at the 1st index.**
* “decrease” **decreases** **all elements** in the array **with 1**.

## Input

On the **first input line** you will be given **the initial array values** separated by a single space.

On the **next lines** you will receive commands **until** you receive the **command “end”**. The **commands are** as follow:

* “swap {index1} {index2}”
* “multiply {index1} {index2}”
* “decrease”

## Output

**The output** should be printed on the console and consist **element** **of the** **modified array** – **separated by “, “**(**comma and single space**).

## Constraints

* Commands will be: “**swap**”, “**multiply**”, “**decrease**” and “**end**”
* **Elements of the array** will be **integer numbers** in the range **[-231...231]**
* **Count of the array elements** will be in the range **[2...100]**
* **Indexes** **will be always in the range of the array**

## Examples

|  |  |  |
| --- | --- | --- |
| **Input** | **Output** | **Comments** |
| 23 -2 321 87 42 90 -123  swap 1 3  swap 3 6  swap 1 0  multiply 1 2  multiply 2 1  decrease  end | 86, 7382, 2369942, -124, 41, 89, -3 | 23 -2 321 87 42 90 -123 – initial values  swap 1(-2) and 3(87) ▼  23 87 321 -2 42 90 -123  swap 3(-2) and 6(-123) ▼  23 87 321 -123 42 90 -2  swap 1(87) and 0(23) ▼  87 23 321 -123 42 90 -2  multiply 1(23) 2(321) = 7383 ▼  87 7383 321 -123 42 290 -2  multiply 2(321) 1(7383) = 2369943 ▼  87 7383 2369943 -123 42 90 -2  decrease – all - 1 ▼  86 7383 2369942 -124 41 89 -3 |
| 1 2 3 4  swap 0 1  swap 1 2  swap 2 3  multiply 1 2  decrease  end | 1, 11, 3, 0 |  |

|  |  |  |
| --- | --- | --- |
| **Input JavaScript** | **Output** | **Comments** |
| [  '23 -2 321 87 42 90 -123',  'swap 1 3',  'swap 3 6',  'swap 1 0',  'multiply 1 2',  'multiply 2 1',  'decrease',  'end'  ] | 86, 7382, 2369942, -124, 41, 89, -3 | 23 -2 321 87 42 90 -123 – initial values  swap 1(-2) and 3(87) ▼  23 87 321 -2 42 90 -123  swap 3(-2) and 6(-123) ▼  23 87 321 -123 42 90 -2  swap 1(87) and 0(23) ▼  87 23 321 -123 42 90 -2  multiply 1(23) 2(321) = 7383 ▼  87 7383 321 -123 42 290 -2  multiply 2(321) 1(7383) = 2369943 ▼  87 7383 2369943 -123 42 90 -2  decrease – all - 1 ▼  86 7383 2369942 -124 41 89 -3 |
| [  '1 2 3 4',  'swap 0 1',  'swap 1 2',  'swap 2 3',  'multiply 1 2',  'decrease',  'end'  ] | 1, 11, 3, 0 |  |

import java.util.Scanner;  
  
public class Main {  
 public static void main(String[] args) {  
 Scanner scanner = new Scanner(System.*in*);  
  
 String[] numbersAsText = scanner.nextLine().split(" ");  
 int[] numbers = new int[numbersAsText.length];  
 for (int i = 0; i < numbersAsText.length; i++) {  
 numbers[i] = Integer.*parseInt*(numbersAsText[i]);  
 }  
  
 String input = scanner.nextLine();  
 while (!"end".equals(input)) {  
 String[] tokens = input.split(" ");  
 String command = tokens[0];  
 switch (command) {  
 case "swap":  
 int indexOne = Integer.*parseInt*(tokens[1]);  
 int indexTwo = Integer.*parseInt*(tokens[2]);  
 int temp = numbers[indexTwo];  
 numbers[indexTwo] = numbers[indexOne];  
 numbers[indexOne] = temp;  
 break;  
 case "multiply":  
 int multiplyOne = Integer.*parseInt*(tokens[1]);  
 int multiplyTwo = Integer.*parseInt*(tokens[2]);  
 int product = numbers[multiplyOne] \* numbers[multiplyTwo];  
 numbers[multiplyOne] = product;  
 break;  
 case "decrease":  
 for (int i = 0; i < numbers.length; i++) {  
 numbers[i] -= 1;  
 }  
 break;  
 }  
 input = scanner.nextLine();  
 }  
  
 for (int i = 0; i < numbers.length-1; i++) {  
 System.*out*.print (numbers[i] + ", ");  
 }  
  
 System.*out*.print(numbers[numbers.length-1]);  
 }  
}