# Angry Pet

*Tom is very angry with his owner, because he left him alone during the teamwork defenses for the Programming Fundamentals Course at SoftUni. It’s time for Tom to get his payback and he will do it, by breaking various household items.*

Each item has a **price rating** which is a number that describes how valuable is that item for Tom’s owner. You will be **given an entry point** from which Tom will start **breaking** the **items to his left**, and **then** to **his right**. Tom **will never** break the item at the **entry point**.

You must calculate the damage to **both** his **left**, and **right**, then print **only the higher (bigger)** damage to the household. If both **sums** are **equal** print the **left** one.

## Input / Constrains

* On the **first line** you will receive the **price ratings**, separated by (space). Each element will be integer in range **[-231… 231]**
* On the **second line** you will receive the **entry point**, which will always be between the second and the penultimate element in the array
* On the **third line** you will receive the **type of items** Tom wants to break, which will be one of the following:
  + **cheap** – items that have **lower price rating** than the entry point item
  + **expensive** – items that have **same price rating**, or **higher price rating** than the entry point item
* On the **last line** you will receive the **type of price ratings** that Tom will look for, which will be one of the following:
  + **positive** – price ratings **above** 0
  + **negative** – price ratings **below** 0
  + **all** – **any** price ratings

## Output

* Single line containing the sum of price ratings and their position based on the entry point in the following format:
  + "{**position**} – {**sum of price ratings**}"

## Examples

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Input** | **Output** |  | **Input** | **Output** |
| 1 5 1  1  cheap  all | Left - 1 |  | -2 2 1 5 9 3 2 -2 1 -1 -3 3  7  expensive  positive | Left - 22 |

## JS Input

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Input** | **Output** |  | **Input** | **Output** |
| ([1, 5, 1],  1,  "cheap",  "all") | Left - 1 |  | ([-2, 2, 1, 5, 9, 3, 2, -2, 1, -1, -3, 3],  7,  "expensive",  "positive") | Left - 22 |