

Problem 2. Listmon says

Input / Constraints

We are at the Listmon's playground. The collection defines its own rules and if we want to beat It in its own game, we must play by the rules. A game is pretty similar to 'Simon says'. So, It will tell us what to do and we should read very carefully what It wants from us.

At the beginning of the game Listmon will give us an input with different elements separated by space (**one or several**). The elements will be numbers.

After that It will start giving us commands in format:

- **set**
- **filter {command} (where command will be either odd or even)**
- **multiply {number}**
- **divide {number}**
- **slice {indexN} {indexM}**
- **sort**
- **reverse**

*If you receive command 'set' you should check if the list is not **already** with unique elements, if this is so - print:

- "It is a set"

If it isn't, make it one and print it **as a list**. Keep **the order of the elements exactly** the **same** as Listmon gave it to you, just remove the non-unique elements.

*If you receive 'filter' you should print only **either odds elements, or even**, depending on the command next to filter.

If there are no elements in the list after filter command **do not print the list**.

*If you receive 'multiply', multiply every element of the list by the given number.

*If you receive 'divide' you must **divide every element by the given number and print the list**. If the number is 0, print:

- 'ZeroDivisionError caught' And **do not print the list**.

*If you receive 'slice' {indexN} {indexM} you should print the elements **from n to m including** without actually changing the list! Keep in mind that Listmon is tricky and can give you **indexes which are not part of the list**. In this case you should print:

- 'IndexError caught'.

And **do not print the list**.

Index N always will be smaller number than M if the two indexes are valid

*If you receive **sort** - print the elements in ascending order.

*If you receive **reverse** – print all elements in reversed order.

ALWAYS keep in mind that the **Listmon's list never should be changed**, otherwise we are going to lose the game.

NOTE:

After each command you should print the result **ONLY if there are elements in the list**. If the list is **empty do not print it!**

Output

When you receive a command which says **'exhausted'**, you should print the count of rounds played in format:

- "I beat It for {count} rounds!"

where *count* is the number of commands you have received during the game.

Examples

Input	Output	Comments
1 3 2 4 5 set slice 1 5 sort reverse exhausted	It is a set IndexError caught [1, 2, 3, 4, 5] [5, 4, 2, 3, 1] I beat It for 4 rounds!	On the first line we receive Listmon's list. On the second we see set. We check if all elements are unique – they are so we just print 'It is a set'. After that we receive 'slice 1 5' the first index is valid but the second one is not because we have only 4 indexes (5 elements), so we print 'IndexError caught'. After that we receive 'sort' and we sort all elements by ascending order and print them. After that we receive 'reverse' and we should reverse Listmon's list , so we get [5, 4, 2, 3, 1] not [5, 4, 3, 2, 1], because we never ever change the Listmon's list. After command 'exhausted' we print 'I beat It for 4 rounds!' because in this case we receive 4 commands from It.

Input	Output	Comments
1 15 3 279 12 filter odd multiply 7 divide 0 reverse exhausted	[1, 15, 3, 279] [7, 105, 21, 1953, 84] ZeroDivisionError caught [12, 279, 3, 15, 1] I beat It for 4 rounds!	