

# Introduction to Regex Module

## re

A regular expression (or RegEx) specifies a set of strings that matches it.

A **regex** is a sequence of characters that defines a search pattern, mainly for the use of string pattern matching.

The **re.search()** expression scans through a string looking for the *first* location where the regex pattern produces a match.  
It either returns a **MatchObject** instance or returns **None** if no position in the string matches the pattern.

### Code

```
>>> import re
>>> print bool(re.search(r"ly", "similarly"))
True
```

The **re.match()** expression only matches at the *beginning* of the string.  
It either returns a **MatchObject** instance or returns **None** if the string does not match the pattern.

### Code

```
>>> import re
>>> print bool(re.match(r"ly", "similarly"))
False
>>> print bool(re.match(r"ly", "ly should be in the beginning"))
True
```

## Task

You are given a string  $N$ .  
Your task is to verify that  $N$  is a floating point number.

In this task, a valid float number must satisfy *all* of the following requirements:

> Number can start with **+**, **-** or **.** symbol.

For example:

✓ +4.50

✓ -1.0

✓ .5

✓ -.7

✓ +.4

✗ **++4.5**

> Number must contain *at least* 1 decimal value.

For example:

✗ **12.**

✓ 12.0

- > Number must have exactly one `.` symbol.
- > Number must not give any exceptions when converted using *`float(N)`*.

### Input Format

The first line contains an integer  $T$ , the number of test cases.  
The next  $T$  line(s) contains a string  $N$ .

### Constraints

$$0 < T < 10$$

### Output Format

Output *True* or *False* for each test case.

### Sample Input

```
4
4.000
-1.00
+4.54
SomeRandomStuff
```

### Sample Output

```
False
True
True
False
```

### Explanation

**4.000**:  $O$  is not a digit.  
**−1.00**: is valid.  
**+4.54**: is valid.  
*SomeRandomStuff*: is not a number.