**JAVA PROGRAM**

**TASK 1:**

Given a string  **A**, print Yes if it is a palindrome, print No otherwise.

**Constraints**

* **A** will consist at most  **50** lower case english letters.

**Sample Input**

madam

**Sample Output**

Yes

**TASK 2:**

Using **Regex**, we can easily match or search for patterns in a text. Before searching for a pattern, we have to specify one using some well-defined syntax.

In this problem, you are given a pattern. You have to check whether the syntax of the given pattern is valid.

**Note**: In this problem, a regex is only valid if you can compile it using the [Pattern.compile](http://docs.oracle.com/javase/6/docs/api/java/util/regex/Pattern.html#compile%28java.lang.String%29) method.

**Input Format**

The first line of input contains an integer **N** , denoting the number of test cases. The next **N** lines contain a string of any printable characters representing the pattern of a regex.

**Output Format**

For each test case, print Valid if the syntax of the given pattern is correct. Otherwise, print Invalid. Do not print the quotes.

**Sample Input**

3

([A-Z])(.+)

[AZ[a-z](a-z)

batcatpat(nat

**Sample Output**

Valid

Invalid

Invalid

**TASK 3:**

You are given a list of student information: ID, FirstName, and CGPA. Your task is to rearrange them according to their CGPA in decreasing order. If two student have the same CGPA, then arrange them according to their first name in alphabetical order. If those two students also have the same first name, then order them according to their ID. No two students have the same ID.

**Hint**: You can use comparators to sort a list of objects. See the [oracle docs](http://docs.oracle.com/javase/tutorial/collections/interfaces/order.html) to learn about comparators.

**Input Format**

The first line of input contains an integer **N**, representing the total number of students. The next **N** lines contains a list of student information in the following structure:

ID Name CGPA

**Constraints**

* **2<=N<=1000**
* **0<=ID<=100000**
* **5<=|NAME|<=30**
* **0<=CGPA<=4.00**

The name contains only lowercase English letters. The  **ID** contains only integer numbers without leading zeros. The CGPA will contain, at most, 2 digits after the decimal point.

**Output Format**

After rearranging the students according to the above rules, print the first name of each student on a separate line.

**Sample Input**

5

33 Rumpa 3.68

85 Ashis 3.85

56 Samiha 3.75

19 Samara 3.75

22 Fahim 3.76

**Sample Output**

Ashis

Fahim

Samara

Samiha

Rumpa

-----------------------------------------\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*----------------------------------