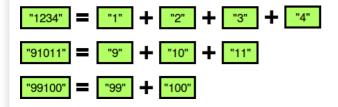# Separate the Numbers ★

Problem | Submissions | Leaderboard | Discussions | Editorial 🔒

A numeric string, s, is beautiful if it can be split into a sequence of two or more positive integers, $a[1], a[2], \ldots, a[n]$, satisfying the following conditions:

1. $a[i] - a[i-1] = 1$ for any $1 < i \le n$ (i.e., each element in the sequence is 1 more than the previous element).
2. No $a[i]$ contains a leading zero. For example, we can split $s = 10203$ into the sequence $\{1, 02, 03\}$, but it is not beautiful because 02 and 03 have leading zeroes.
3. The contents of the sequence cannot be rearranged. For example, we can split $s = 312$ into the sequence $\{3, 1, 2\}$, but it is not beautiful because it breaks our first constraint (i.e., $1 - 3 \ne 1$).

The diagram below depicts some beautiful strings:

"1234" = "1" + "2" + "3" + "4"

"91011" = "9" + "10" + "11"

"99100" = "99" + "100"

Perform q queries where each query consists of some integer string s. For each query, print whether or not the string is beautiful on a new line. If it is beautiful, print YES x, where x is the first number of the increasing sequence. If there are multiple such values of x, choose the smallest. Otherwise, print NO.

Function Description

Complete the separateNumbers function in the editor below.

separateNumbers has the following parameter:

- s: an integer value represented as a string

Prints

- string: Print a string as described above. Return nothing.

Input Format

The first line contains an integer q, the number of strings to evaluate.

Each of the next q lines contains an integer string s to query.

Constraints

- $1 \le q \le 10$
- $1 \le |s| \le 32$
- $s[i] \in [0-9]$

Author | DmitriyH
Difficulty | Easy
Max Score | 100
Submitted By | 5388

NEED HELP?
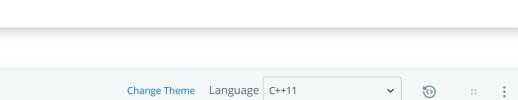
🔲 View discussions
📖 View editorial
🏆 View top submissions

RATE THIS CHALLENGE

★ ★ ★ ★ ★

MORE DETAILS

⬇ Download problem statement
⬇ Download sample test cases
✎ Suggest Edits

f ⮑ in

Change Theme   Language C++11

```cpp
1   #include <bits/stdc++.h>
2
3   using namespace std;
4
5   string ltrim(const string &);
6   string rtrim(const string &);
7
8   /*
9    * Complete the 'separateNumbers' function below.
10   *
11   * The function accepts STRING s as parameter.
```

```
12      */
13
14      void separateNumbers(string s) {
15
16      }
17
18      int main()
19      {
20          string q_temp;
21          getline(cin, q_temp);
22
23          int q = stoi(ltrim(rtrim(q_temp)));
24
25          for (int q_itr = 0; q_itr < q; q_itr++) {
26              string s;
27              getline(cin, s);
28
29              separateNumbers(s);
```

Line: 56 Col: 1

⬆ Upload Code as File   ☐ Test against custom input   Run Code   Submit Code