**Sum of all substrings of a number**

Show Topic Tags   

Given a integer represented as a string, we need to get the sum of all possible sub-strings of this string.

Examples:

Input : N = “1234”

Output : 1670

Sum = 1 + 2 + 3 + 4 + 12 + 23 +

34 + 123 + 234 + 1234

= 1670

Input : N = “421”

Output : 491

Sum = 4 + 2 + 1 + 42 + 21 + 421

= 491

**Input:**  
The first line of input contains an integer T denoting the no of test cases. Then T test cases follow. Each test case contains an integer N.  
  
**Output:**  
For each test case in a new line print the required result.  
  
**Constraints:**  
1<=T<=200  
1<=N<=10^5+6  
  
**Example:  
Input:**  
2  
1234  
421  
**Output:**  
1670  
491

\*\*For More Examples Use Expected Output\*\*

<http://practice.geeksforgeeks.org/problems/sum-of-all-substrings-of-a-number/0>

/\*

\* To change this template, choose Tools | Templates

\* and open the template in the editor.

\*/

package javaapplication248;

import java.io.BufferedReader;

import java.io.IOException;

import java.io.InputStreamReader;

import java.util.ArrayList;

import java.util.Arrays;

/\*\*

\*

\* @author Administrador

\*/

public class JavaApplication248 {

public static void main(String[] args) throws IOException {

// TODO code application logic here

BufferedReader br = new BufferedReader(new InputStreamReader(System.in));

int t = Integer.parseInt(br.readLine());

while(t-- > 0) {

String n = br.readLine().trim();

int sum =0;

for(int i =0; i<n.length(); i++) {

String concat = "";

for(int j=i; j<n.length(); j++) {

concat += n.charAt(j);

//System.out.print(concat + " ");

sum += Integer.parseInt(concat);

}

}

System.out.println(sum);

}

}

}