**Repeated sum of digits**

Show Topic Tags   

Given an integer N, recursively sum digits of N until we get a single digit.  The process can be described below

If N < 10

digSum(N) = N

Else

digSum(N) = Sum(digSum(N))

Example:

Input : 1234

Output : 1

Explanation : The sum of 1+2+3+4 = 10,

digSum(x) == 10

Hence ans will be 1+0 = 1

**Input:**

The first line contains an integer T, total number of test cases. Then following T lines contains an integer N.

**Output:**

Repeated sum of digits of N.

**Constraints:**

1 ≤ T ≤ 100

1 ≤ N ≤ 1000000

**Example:**

Input  
2  
123  
9999

Output  
6  
9

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

<http://practice.geeksforgeeks.org/problems/repeated-sum-of-digits/0>

/\*

\* To change this template, choose Tools | Templates

\* and open the template in the editor.

\*/

package javaapplication244;

import java.io.BufferedReader;

import java.io.IOException;

import java.io.InputStreamReader;

import java.util.ArrayList;

import java.util.Arrays;

import java.util.Collections;

import java.util.HashMap;

import java.util.List;

/\*\*

\*

\* @author Administrador

\*/

public class JavaApplication244 {

/\*\*

\* @param args the command line arguments

\*/

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) {

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

int sum = 0;

int copia = n;

do

{

sum = 0;

while (copia > 0)

{

sum += copia % 10;

copia /= 10;

}

copia = sum;

} while (String.valueOf(sum).length() != 1);

System.out.println(sum);

}

}

}