**Keypad typing**

[string](http://www.practice.geeksforgeeks.org/tag-page.php?tag=string&isCmp=0)

[Amazon](http://www.practice.geeksforgeeks.org/tag-page.php?tag=Amazon&isCmp=1)

You are given N strings of alphabet characters and the task is to find their matching decimal representation as on the shown keypad. Output the decimal representation corresponding to the string. For ex: if you are given “amazon” then its corresponding decimal representation will be 262966.



**Input:**

The first line of input contains an integer T denoting the number of test cases. Then T test cases follow. Each test case consists of a single line containing a string.

**Output:**

For each test case, print in a new line, the corresponding decimal representation of the given string.

**Constraints:**

1 ≤ T ≤ 100  
1 ≤ length of String ≤ 100  
  
**Example:**

**Input**  
2  
geeksforgeeks  
geeksquiz

**Output**  
4335736743357  
433577849

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

<http://www.practice.geeksforgeeks.org/problem-page.php?pid=871>

#include <iostream>

#include <stdio.h>

#include <set>

#include <math.h>

#include <map>

#define ll long long int

using namespace std;

int main() {

int t;

scanf("%d", &t);

while(t--) {

std::string s;

cin >> s;

std::map<char, int> m;

m['A'] = 2;

m['B'] = 2;

m['C'] = 2;

m['D'] = 3;

m['E'] = 3;

m['F'] = 3;

m['G'] = 4;

m['H'] = 4;

m['I'] = 4;

m['J'] = 5;

m['K'] = 5;

m['L'] = 5;

m['M'] = 6;

m['N'] = 6;

m['O'] = 6;

m['P'] = 7;

m['Q'] = 7;

m['R'] = 7;

m['S'] = 7;

m['T'] = 8;

m['U'] = 8;

m['V'] = 8;

m['W'] = 9;

m['X'] = 9;

m['Y'] = 9;

m['Z'] = 9;

m['a'] = 2;

m['b'] = 2;

m['c'] = 2;

m['d'] = 3;

m['e'] = 3;

m['f'] = 3;

m['g'] = 4;

m['h'] = 4;

m['i'] = 4;

m['j'] = 5;

m['k'] = 5;

m['l'] = 5;

m['m'] = 6;

m['n'] = 6;

m['o'] = 6;

m['p'] = 7;

m['q'] = 7;

m['r'] = 7;

m['s'] = 7;

m['t'] = 8;

m['u'] = 8;

m['v'] = 8;

m['w'] = 9;

m['x'] = 9;

m['y'] = 9;

m['z'] = 9;

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

cout << m[s[i]];

}

cout << endl;

}

//system("pause");

return 0;

}