**Magical Word**

Attempted by: **5381**

/

Accuracy: **41%**

/

Maximum Score: **30**

/

155 Votes

Tag(s):

Easy-Medium

**PROBLEM**

**EDITORIAL**

**MY SUBMISSIONS**

**ANALYTICS**

Dhananjay has recently learned about [ASCII](http://www.asciitable.com/) values.He is very fond of experimenting. With his knowledge of [ASCII](http://www.asciitable.com/) values and character he has developed a special word and named it Dhananjay's Magical word.

A word which consist of alphabets whose [ASCII](http://www.asciitable.com/) values is a prime number is an Dhananjay's Magical word. An alphabet is Dhananjay's Magical alphabet if its [ASCII](http://www.asciitable.com/) value is prime.

Dhananjay's nature is to boast about the things he know or have learnt about. So just to defame his friends he gives few string to his friends and ask them to convert it to Dhananjay's Magical word. None of his friends would like to get insulted. Help them to convert the given strings to Dhananjay's Magical Word.

**Rules for converting:**

1.Each character should be replaced by the nearest Dhananjay's Magical alphabet.

2.If the character is equidistant with 2 Magical alphabets. The one with lower [ASCII](http://www.asciitable.com/) value will be considered as its replacement.

**Input format:**

First line of input contains an integer T number of test cases. Each test case contains an integer N(denoting the length of the string) and a string S.

**Output Format:**

For each test case, print Dhananjay's Magical Word in a new line.

**Constraints:**

1 <= T <= 100

1 <= |S| <= 500

**SAMPLE INPUT**

1

6

AFREEN

**SAMPLE OUTPUT**

CGSCCO

**Explanation**

ASCII values of alphabets in AFREEN are 65, 70, 82, 69 ,69 and 78 respectively which are converted to CGSCCO with ASCII values 67, 71, 83, 67, 67, 79 respectively. All such ASCII values are prime numbers.

**Time Limit:**0.5 sec(s) for each input file.

**Memory Limit:**256 MB

**Source Limit:**1024 KB

**Marking Scheme:**Marks are awarded when all the testcases pass.

**Allowed Languages:**C, C++, Clojure, C#, D, Erlang, F#, Go, Groovy, Haskell, Java, Java 8, JavaScript(Rhino), JavaScript(Node.js), Lisp, Lisp (SBCL), Lua, Objective-C, OCaml, Octave, Pascal, Perl, PHP, Python, Python 3, R(RScript), Racket, Ruby, Rust, Scala 2.11.8, Swift, Visual Basic

<https://www.hackerearth.com/practice/basic-programming/input-output/basics-of-input-output/practice-problems/algorithm/magical-word/>

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.IO;

namespace ConsoleApplication2

{

class Program

{

static void Main(string[] args)

{

int[] primos = /\*{ 2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59,\*/ { 67, 71, 73, 79, 83, 89, 97, 101, 103, 107, 109, 113 }; //, 131, 137, 139, 149, 151, 157, 163, 167, 173, 179, 181, 191, 193, 197, 199, 211, 223, 227, 229, 233, 239, 241, 251, 257, 263, 269, 271, 277, 281, 283, 293, 307, 311, 313, 317, 331, 337, 347, 349, 353, 359, 367, 373, 379, 383, 389, 397, 401, 409, 419, 421, 431, 433, 439, 443, 449, 457, 461, 463, 467, 479, 487, 491, 499 };

int t = int.Parse(Console.ReadLine());

while (t-- > 0)

{

int n = int.Parse(Console.ReadLine());

string s = Console.ReadLine();

// string s = "1234567890";

// string s = "S@#!#$op";

//string s = "AvfmaLgLRpQadLyThsxVzkUqbFOdxfbLGdpBWOwmAnflENlYFbdhNHerHVtZkaPLgMtNQovVHpwGfHJdXXWAhYrhwXKPxtnpxCIsaXVAkcxTpVprFNeOVcnSEsgIvfqXPRSUASSDCvAGrFJCDbzGLFhrMYWALElChmurLrEeQttIWctyhQXzZUVAYuCIZecBJbXMxlMHFbZxJRTSZJmZAwCggGabVsovqBrdmmbCTaIHDfUunLFntfGzodKqoAKwCassKMDybethRaQgegsOawfNCNrIAkECEKpbwElhvWtlZBEZqJQpEkzpiSjrqZZIHbszUxwuWreXkxFKxSAiKoemIqETGVxcCjweKhbyXxhVKCbNTQBKgHD";

//string s = "1234567890";

//10

//1234567890

//8

//S@#!#$op

//6

//S#!v@m

//11

//!@#$%^&\*()\_

//15

//+=-\|]}[{'";:><

// string s = "AWGrOwUcFWnghtAyvxSb";

//CYGqOqSaGYmggqCqqqSa

bool[] nums = new bool[200];

for (int i = 0; i < primos.Length; i++)

{

nums[primos[i]] = true;

}

//for (int i = 0; i < nums.Length; i++)

//{

// Console.WriteLine(i + " " + nums[i] + " ");

//}

//string s = "AFREEN";

string ans = "";

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

{

//Console.Write((int)s[i] + " ");

int ASCIant = 0, ASCIpost = 0;

int indice = s[i];

//if (char.IsNumber(s[i]))

//{

// indice = int.Parse(s[i].ToString());

//}

int j = indice;

while (j >= 0)

{

if (nums[j])

{

ASCIant = j;

break;

}

j--;

}

if (j < 0)

{

ASCIant = int.MaxValue;

}

j = indice;

while (j < nums.Length)

{

if (nums[j])

{

ASCIpost = j;

break;

}

j++;

}

int dif\_a = Math.Abs(s[i] - ASCIant);

int dif\_b = Math.Abs(s[i] - ASCIpost);

//Console.WriteLine(((int)(ASCIant) + " " + (int)s[i] + " "

// + ((int)(ASCIpost))));

if (dif\_a <= dif\_b)

{

ans += (char)ASCIant;

}

else if (dif\_b < dif\_a)

{

ans += (char)ASCIpost;

}

}

Console.WriteLine(ans);

//for (int i = 0; i < ans.Length; i++)

//{

// Console.Write((int)ans[i] + " ");

//}

}

Console.ReadLine();

}

}

}