Consider that the English alphabet contains 26 characters, while telephones only have ten digits on the keypad. The letters are mapped onto the digits as shown below:



When you press a digit, the corresponding letter appears on the screen. If you keep pressing the button without pauses, the letters mapped onto the button change in sequence. Let's assume that no button is pressed repeatedly more times than there are letters mapped onto it.

Given the list of digits pressed, return the text that should appear on the screen.

**Example**

* For keys = "44 444", the output should be reverse\_t9(keys) = "hi".
* For keys = "999337777", the output should be reverse\_t9(keys) = "yes".

*Check out [this](https://code.google.com/codejam/contest/351101/dashboard" \l "s=p2) challenge for reference.*

**Input/Output**

* **[time limit] 3000ms (cs)**
* **[input] string keys**

Sequence of key presses, a string of digits (all but '1') and whitespace characters, where a whitespace character means a pause. It is guaranteed that there are no two consecutive '0's or ' 's in the string.

*Constraints:* 1 ≤ keys.length ≤ 1500.

* **[output] string**

The text as a string of lowercase English letters.

<https://codefights.com/challenge/JGNL3YRDqC7By3fbu/main>

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace ConsoleApplication1

{

class Program

{

static string reverse\_t9(string keys)

{

List<string> lista = new List<string>();

int i = 0;

while (i < keys.Length)

{

int cont = 1;

if (keys[i] != ' ')

{

while (i + 1 < keys.Length && keys[i] == keys[i + 1])

{

i++;

cont++;

}

if (keys[i] == '7' || keys[i] == '9')

{

if (cont > 4)

{

cont = cont % 4 + 1;

}

}

else

{

if (cont > 3)

{

cont = cont % 3 + 1;

}

}

lista.Add(keys[i] + " " + cont);

}

i++;

}

//foreach (string s in lista)

//{

// Console.WriteLine(s);

//}

var diccio = new Dictionary<string, char>();

diccio["2 1"] = 'a';

diccio["2 2"] = 'b';

diccio["2 3"] = 'c';

diccio["3 1"] = 'd';

diccio["3 2"] = 'e';

diccio["3 3"] = 'f';

diccio["4 1"] = 'g';

diccio["4 2"] = 'h';

diccio["4 3"] = 'i';

diccio["5 1"] = 'j';

diccio["5 2"] = 'k';

diccio["5 3"] = 'l';

diccio["6 1"] = 'm';

diccio["6 2"] = 'n';

diccio["6 3"] = 'o';

diccio["7 1"] = 'p';

diccio["7 2"] = 'q';

diccio["7 3"] = 'r';

diccio["7 4"] = 's';

diccio["8 1"] = 't';

diccio["8 2"] = 'u';

diccio["8 3"] = 'v';

diccio["9 1"] = 'w';

diccio["9 2"] = 'x';

diccio["9 3"] = 'y';

diccio["9 4"] = 'z';

diccio["0 1"] = ' ';

// Console.WriteLine();

string ans = "";

foreach (string elem in lista)

{

//Console.Write(diccio[elem]);

ans += diccio[elem];

}

return ans;

}

static void Main(string[] args)

{

//string keys = "99933 7777 444 44";

//string keys = "44204420442";

// string keys = "444444444444555555";

//string keys = "2 22 2223 33 3334 44 4445 55 5556 66 666";

string keys = "999337777";

Console.WriteLine(reverse\_t9(keys));

Console.ReadLine();

}

}

}