**Benford's law**

Yesterday Vasya learned about the amazing Benford's law and wants to test its operation. To do this, he took a text with up-to-date data on the tallest buildings in the world and wants to get statistics: how many times each digit stands in the place of the most significant digit in the numbers from his text.

The GetBenfordStatistics method should return an array of numbers, in which the i-th position contains the statistics for digit i.



**Tallest buildings in the world as of 2020**

Burj Khalifa 830

Petronius (oil platform) 640

Tokyo Skytree 634

KVLY-TV mast 629

Canton Tower 604

Abraj Al Bait Towers 601

Bullwinkle (oil platform) 529

Troll A platform 472

Lualualei VLF transmitter 458

Petronas Twin Towers 452

Willis Tower 442

Ekibastuz GRES-Two Power Station 420

Dimona Radar Facility 400

Kiev TV Tower 385

Zhoushan Island Overhead Powerline Tie 370

Gerbrandy Tower 367

TV Tower Vinnytsia 354

Millau Viaduct 342

Amazon Tall Tower Observatory 325

Lakihegy Tower 314

Jinping-I Dam 305

Star Tower 291

H-One Tower 274

Djamaa el Djazaïr 265

Mohammed bin Rashid Al Maktoum Solar Park 262

LR 248

GE wind turbine at Naturstromspeicher Gaildorf 247

Statue of Unity 240

Noble Lloyd Noble 214

Kalisindh Thermal Power Station 202

Gateway Arch 192

Main tower of Kuwait Towers 187

Anaconda Smelter Stack 178

Olympic Stadium 175

San Jacinto Monument 174

Niederaussem Power Station 172

Jeddah Flagpole 171

High Roller 168

Mole Antonelliana 168

Ulmer Münster 162

Vehicle Assembly Building 160

Santa Cruz del Valle de los Caídos 152

Arecibo Telescope 150

Kingda Ka 139

Great Pyramid of Giza 139

Kuala Lumpur International Airport Two Control Tower 141

Zumanjaro: Drop of Doom 139

Kockums Crane 138

Jetavanaramaya 122

Gliwice Radio Tower 118

Gasometer Oberhausen 118

Schapfen Mill Tower 115

Pillar of third section of Gletscherbahn Kaprun 114

Joseph Chamberlain Memorial Clock Tower 100

Éole 96

Mjøstårnet 85

Ericsson Globe 85

Île Vierge Lighthouse 83

Murudeshwara Temple 76

Source: [wikipedia](https://en.wikipedia.org/wiki/List_of_tallest_buildings_and_structures#Tallest_structure_by_category)

public static void Main()

{

PrintNumbers(GetBenfordStatistics("1"));

PrintNumbers(GetBenfordStatistics("abc"));

PrintNumbers(GetBenfordStatistics("123 456 789"));

PrintNumbers(GetBenfordStatistics("abc 123 def 456 gf 789 i"));

PrintNumbers(GetBenfordStatistics(tallestBuildings));

}

public static int[] GetBenfordStatistics(string text)

{

var statistics = new int[10];

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

{

//...

}

return statistics;

}

**Hints:**

Loop through the text and find all the first digits of the numbers encountered.

To understand that a character is a digit use the char.IsDigit () method

The simplest way to get a digit from a character is to subtract the '0' from the character. The point is that char is a numeric data type that you can perform arithmetic operations on.

A character is the first digit of a number if char.IsDigit returns true on it, and false on the previous character (if any).

**Code:**

using System;

using System.Collections.Generic;

using System.IO;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace umop6o7BenfordsLaw

{

class Program

{

public static void Main()

{

string tallestBuildings = File.ReadAllText("tallestB.txt");

PrintNumbers(GetBenfordStatistics("1"));

PrintNumbers(GetBenfordStatistics("abc"));

PrintNumbers(GetBenfordStatistics("123 456 789"));

PrintNumbers(GetBenfordStatistics("abc 123 def 456 gf 789 i"));

PrintNumbers(GetBenfordStatistics(tallestBuildings));

Console.ReadKey();

}

public static void PrintNumbers(int[] statistics)

{

foreach (var e in statistics)

Console.Write(e+" ");

Console.WriteLine();

Console.WriteLine();

}

public static int[] GetBenfordStatistics(string text)

{

List<int> digits = new List<int>();

var statistics = new int[10];

if (char.IsDigit(text[0]))

{

digits.Add(text[0] - '0');

}

for (int i = 1; i < text.Length; i++)

{

if (char.IsDigit(text[i]) & !char.IsDigit(text[i - 1]))

{

digits.Add(text[i]-'0');

}

}

foreach (var e in digits)

Console.Write(e+" ");

Console.WriteLine();

for (int j = 0; j < 10; j++)

{

for (int i = 0; i < digits.Count; i++)

{

if (j+1== digits[i])

{

statistics[j]++;

}

}

}

return statistics;

}

}

}