**1. uzdevums: Tabula izveide ar metodes un funkciju palīdzību**

Izveidojiet C# konsoles lietojumprogrammu.

Uzrakstiet **metodi** ar nosaukumu "**TabulasLinija**", kas saņem divus parametrus:

1. svītriņu skaits,
2. reižu skaits.

Piemēram :

\*---\*---\*---\*---\*

1- parametrs saņem vērtību-> 3 (trīs svītriņas starp zvaigznēm),

2- parametrs->4 ( 4x izpildās)

Uzrakstiet **metodi** ar nosaukumu "**TabulasRinda**", kas saņem 4. parametrus:

1. Rindas dati (1D masīvs),
2. Max vērtību
3. Min vertibu,
4. Sum vērtību

…un izdrukā (piemēram)

| 1 | 2 | 3 | 4 | max | min | sum |

Piemēram :

Int[] dati={1,2,3,4}

Uzrakstiet **funkciju** ar nosaukumu "**TabulasDati**", kas saņem divus parametrus:

1. vārdu
2. uzvārdu

Pēc šiem datiem **atgriež** 2D masīvu, kas uzģenerē datus ar **gadījuma** skaitļiem robežās no **vārda garuma** līdz **varda+uzvarda garumam (ieskaitot)**

Uzrakstiet **metodi** ar nosaukumu "**TabulaGalva**", kas saņem parametru:

1. Vārds

Izmantojot metodes **TabulasLinija** un **TabulasRinda** izvada tabulas galvā **vārdu un Min, Max, Sum**

Uzrakstiet **metodi** ar nosaukumu "**Tabula**", kas saņem parametru:

1. **2D masīvu**

Izmantojot metodes **TabulasLinija** un **TabulasRinda** izvada tabulas galvā **rindas datus un Min, Max, Sum vērtības**

Uzrakstiet **funkciju** ar nosaukumu "**TabulasMaina**", kas saņem parametru:

1. **2D masīvu**

Funkcija izveido 2D masīvu, kas apmaina kolonas ar rindas datiem vietām un atgriež **2D masīvu**

"**Main**" metodē izsaucam:

**TabulasDati…**

**TabulaGalva…**

**Tabula….**

**Console.WriteLine(….)**

**TabulasMaina…**

**TabulaGalva…**

**Tabula….**

using System;

using System.Linq;

class Program

{

static void Main(string[] args)

{

Console.Write("Ievadi vārdu: ");

string ievade1 = Console.ReadLine();

Console.Write("Ievadi uzvārdu: ");

string ievade2 = Console.ReadLine();

int[,] sakumaMasivs = DatuVeidosana(ievade1, ievade2);

GalvenesIzvade(ievade1);

TabulasIzvade(sakumaMasivs);

Console.WriteLine("\n\n\n");

int[,] apgrieztsMasivs = ApgrieztMasivu(sakumaMasivs);

GalvenesIzvade(ievade2);

TabulasIzvade(apgrieztsMasivs);

}

static void LinijasDruka(int simboli, int reizes)

{

for (int k = 0; k < reizes; k++)

{

Console.Write("\*");

for (int m = 0; m < simboli; m++) Console.Write("-");

}

Console.WriteLine("\*");

}

static void RindasDruka(int[] skaitli, int lielakais, int mazakais, int summa)

{

foreach (int sk in skaitli)

{

if (sk < 10)

Console.Write($"| {sk} ");

else

Console.Write($"| {sk} ");

}

Console.Write(lielakais < 10 ? $"| {lielakais} " : $"| {lielakais} ");

Console.Write(mazakais < 10 ? $"| {mazakais} " : $"| {mazakais} ");

if (summa < 10)

Console.Write($"| {summa} ");

else if (summa > 99)

Console.Write($"| {summa} ");

else

Console.Write($"| {summa} ");

Console.WriteLine("|");

}

static int[,] DatuVeidosana(string iev1, string iev2)

{

Random gadijums = new Random();

int min = iev1.Length;

int max = iev1.Length + iev2.Length;

int[,] izklajs = new int[iev2.Length, iev1.Length];

for (int x = 0; x < izklajs.GetLength(0); x++)

{

for (int y = 0; y < izklajs.GetLength(1); y++)

{

izklajs[x, y] = gadijums.Next(min, max + 1);

}

}

return izklajs;

}

static void GalvenesIzvade(string teksts)

{

LinijasDruka(5, teksts.Length + 3);

for (int q = 0; q < teksts.Length; q++)

{

Console.Write($"| {teksts[q]} ");

}

Console.Write("| max ");

Console.Write("| min ");

Console.Write("| sum |");

Console.WriteLine();

}

static void TabulasIzvade(int[,] saturs)

{

int rindas = saturs.GetLength(0);

int kolonnas = saturs.GetLength(1);

int[] vienaRinda = new int[kolonnas];

for (int r = 0; r < rindas; r++)

{

for (int c = 0; c < kolonnas; c++)

{

vienaRinda[c] = saturs[r, c];

}

int max = vienaRinda.Max();

int min = vienaRinda.Min();

int sum = vienaRinda.Sum();

LinijasDruka(5, kolonnas + 3);

RindasDruka(vienaRinda, max, min, sum);

}

LinijasDruka(5, kolonnas + 3);

}

static int[,] ApgrieztMasivu(int[,] ori)

{

int augstums = ori.GetLength(0);

int platums = ori.GetLength(1);

int[,] mainits = new int[platums, augstums];

for (int u = 0; u < augstums; u++)

{

for (int v = 0; v < platums; v++)

{

mainits[v, u] = ori[u, v];

}

}

return mainits;

}

}