# Uzdevums: Tabulas aizpilde izmantojot 2D masīvu

using System;

class Program

{

static void Main()

{

Console.Write("Ievadiet vārdu: ");

string vards = Console.ReadLine();

Console.Write("Ievadiet uzvārdu: ");

string uzvards = Console.ReadLine();

string formattedVards = FormatName(vards, true);

string formattedUzvards = FormatName(uzvards, false);

Console.WriteLine($"\nFormated vards: {formattedVards} (Garums: {vards.Length})");

int min = vards.Length;

int max = vards.Length + uzvards.Length;

Random rand = new Random();

int[,] table = new int[uzvards.Length, vards.Length];

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

{

for (int j = 0; j < vards.Length; j++)

{

table[i, j] = rand.Next(min, max + 1);

}

}

PrintTable(table, formattedVards, formattedUzvards, false);

int[,] reversedTable = new int[vards.Length, uzvards.Length];

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

{

for (int j = 0; j < uzvards.Length; j++)

{

reversedTable[i, j] = table[j, i];

}

}

Console.WriteLine("\nReversed table:");

PrintTable(reversedTable, formattedUzvards.ToLower(), formattedVards.ToLower(), true);

}

static string FormatName(string name, bool fromStart)

{

char[] chars = name.ToLower().ToCharArray();

int start = fromStart ? 1 : chars.Length - 2;

int step = fromStart ? 2 : -2;

for (int i = start; fromStart ? i < chars.Length : i >= 0; i += step)

{

chars[i] = char.ToUpper(chars[i]);

}

return new string(chars);

}

static void PrintTable(int[,] data, string header, string firstColumn, bool isReversed)

{

int rows = data.GetLength(0);

int cols = data.GetLength(1);

Console.Write("+----+");

for (int i = 0; i < cols; i++) Console.Write("---+");

Console.WriteLine(isReversed ? "=====+=====+=====+" : "-----+-----+-----+");

Console.Write("| X |");

foreach (char c in header) Console.Write($" {c} |");

Console.WriteLine(" sum | min | max |");

Console.Write("+----+");

for (int i = 0; i < cols; i++) Console.Write("---+");

Console.WriteLine(isReversed ? "=====+=====+=====+" : "-----+-----+-----+");

for (int i = 0; i < rows; i++)

{

Console.Write($"| {firstColumn[i]} |");

int sum = 0, minVal = int.MaxValue, maxVal = int.MinValue;

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

{

Console.Write($"{data[i, j],3}|");

sum += data[i, j];

minVal = Math.Min(minVal, data[i, j]);

maxVal = Math.Max(maxVal, data[i, j]);

}

Console.Write($"{sum,4} |");

Console.Write($"{minVal,4} |");

Console.Write($"{maxVal,4} |\n");

Console.Write("+----+");

for (int j = 0; j < cols; j++) Console.Write(isReversed ? "===+" : "---+");

Console.WriteLine(isReversed ? "=====+=====+=====+" : "-----+-----+-----+");

}

if (isReversed)

{

Console.Write("|\*sum|");

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

{

int colSum = 0;

for (int i = 0; i < rows; i++) colSum += data[i, j];

Console.Write($"{colSum,3}|");

}

Console.WriteLine(" \* | \* | \* |");

Console.Write("|\*min|");

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

{

int colMin = int.MaxValue;

for (int i = 0; i < rows; i++) colMin = Math.Min(colMin, data[i, j]);

Console.Write($"{colMin,3}|");

}

Console.WriteLine(" \* | \* | \* |");

Console.Write("|\*max|");

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

{

int colMax = int.MinValue;

for (int i = 0; i < rows; i++) colMax = Math.Max(colMax, data[i, j]);

Console.Write($"{colMax,3}|");

}

Console.WriteLine(" \* | \* | \* |");

Console.Write("+===+");

for (int j = 0; j < cols; j++) Console.Write("===+");

Console.WriteLine("=====+=====+=====+");

}

}

}

# Uzdevums- robainie masīvi.

using System;

using System.Text;

class Program

{

static void Main()

{

Console.Write("Ievadiet parolu SKAITU, kas lielaks par 4: ");

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

while (count <= 4)

{

Console.Write("Skaitam jabut lielakam par 4: ");

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

}

Console.Write("Ievadiet GARAKO paroles simbolu skaitu, kas lielaks par 9: ");

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

while (maxLength <= 9)

{

Console.Write("Garumam jabut lielakam par 9: ");

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

}

Console.Write("Vajag Lielos burtus (y/n): ");

bool useUppercase = Console.ReadLine().ToLower() == "y";

Console.Write("Vajag ciparus un simbolus (y/n): ");

bool useSymbols = Console.ReadLine().ToLower() == "y";

string[] passwords = GeneratePasswords(count, maxLength, useUppercase, useSymbols);

Console.WriteLine();

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

{

Console.WriteLine($"{i+1}. parole: {passwords[i]}");

}

}

static string[] GeneratePasswords(int count, int maxLength, bool useUppercase, bool useSymbols)

{

Random rand = new Random();

string[] passwords = new string[count];

string chars = "abcdefghijklmnopqrstuvwxyz";

if (useUppercase) chars += "ABCDEFGHIJKLMNOPQRSTUVWXYZ";

if (useSymbols) chars += "0123456789!@#$%^&\*()\_+-=[]{};':\",./<>?";

for (int i = 0; i < count; i++)

{

int length = rand.Next(9, maxLength + 1);

StringBuilder password = new StringBuilder();

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

{

password.Append(chars[rand.Next(chars.Length)]);

}

passwords[i] = password.ToString();

}

return passwords;

}

}