**Лабораторна робота №1**

**Студент**: Фещенко Микита Вадимович  
**Група**: КН 922в

**Тема лабораторної роботи**

**Знайомство з основами програмування мовою C#**

**Мета роботи**

Навчитися створювати консольні програми мовою програмування C#, освоїти основні операції введення/виведення, типи даних та базові конструкції управління.

﻿/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* Лабораторна робота №1 \*/

/\* Типи даних і введення-виведення \*/

/\* Приклад виконання. Варіант № 14 \*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

using System;

class Program

{

static void Main()

{

// Оголошення змінних

string name1, name2, name3;

uint code1, code2, code3; // Кількість рядків коду

float time1\_sgi1, time1\_sgi2; // Час виконання на SGI Challenge та SGI Indy

// Введення даних для першої програми

Console.Write("1. Введіть: назву програми, кількість рядків коду, час на SGI Challenge, час на SGI Indy > ");

string[] input1 = Console.ReadLine().Split(' ');

name1 = input1[0];

code1 = uint.Parse(input1[1]);

time1\_sgi1 = float.Parse(input1[2]);

time1\_sgi2 = float.Parse(input1[3]);

// Введення даних для другої програми

Console.Write("2. Введіть: назву програми, кількість рядків коду, час на SGI Challenge, час на SGI Indy > ");

string[] input2 = Console.ReadLine().Split(' ');

name2 = input2[0];

code2 = uint.Parse(input2[1]);

time1\_sgi1 = float.Parse(input2[2]);

time1\_sgi2 = float.Parse(input2[3]);

// Введення даних для третьої програми

Console.Write("3. Введіть: назву програми, кількість рядків коду, час на SGI Challenge, час на SGI Indy > ");

string[] input3 = Console.ReadLine().Split(' ');

name3 = input3[0];

code3 = uint.Parse(input3[1]);

time1\_sgi1 = float.Parse(input3[2]);

time1\_sgi2 = float.Parse(input3[3]);

// Виведення таблиці

Console.WriteLine("---------------------------------------------------------------");

Console.WriteLine("| Час виконання деяких програм, які застосовують паралельні алгоритми |");

Console.WriteLine("|-------------------------------------------------------------|");

Console.WriteLine("| Назва програми | Кількість рядків коду | Час SGI Challenge (сек) | Час SGI Indy (сек) |");

Console.WriteLine("|-------------------|-----------------------|------------------------|--------------------|");

// Виведення даних для кожної програми

Console.WriteLine("| {0,-17} | {1,21} | {2,22:F2} | {3,18:F2} |", name1, code1, time1\_sgi1, time1\_sgi2);

Console.WriteLine("| {0,-17} | {1,21} | {2,22:F2} | {3,18:F2} |", name2, code2, time1\_sgi1, time1\_sgi2);

Console.WriteLine("| {0,-17} | {1,21} | {2,22:F2} | {3,18:F2} |", name3, code3, time1\_sgi1, time1\_sgi2);

// Примітка

Console.WriteLine("---------------------------------------------------------------");

}

}  
  
﻿/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* Лабораторна робота №1 \*/

/\* Типи даних і введення-виведення \*/

/\* Приклад виконання. Варіант № 14 \*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

using System;

class Program

{

static void Main()

{

// Оголошення змінних

string name1, name2, name3;

uint code1, code2, code3; // Кількість рядків коду

float time1\_sgi1, time1\_sgi2; // Час виконання на SGI Challenge та SGI Indy

// Введення даних для першої програми

Console.Write("1. Введіть: назву програми, кількість рядків коду, час на SGI Challenge, час на SGI Indy > ");

string[] input1 = Console.ReadLine().Split(' ');

name1 = input1[0];

code1 = uint.Parse(input1[1]);

time1\_sgi1 = float.Parse(input1[2]);

time1\_sgi2 = float.Parse(input1[3]);

// Введення даних для другої програми

Console.Write("2. Введіть: назву програми, кількість рядків коду, час на SGI Challenge, час на SGI Indy > ");

string[] input2 = Console.ReadLine().Split(' ');

name2 = input2[0];

code2 = uint.Parse(input2[1]);

time1\_sgi1 = float.Parse(input2[2]);

time1\_sgi2 = float.Parse(input2[3]);

// Введення даних для третьої програми

Console.Write("3. Введіть: назву програми, кількість рядків коду, час на SGI Challenge, час на SGI Indy > ");

string[] input3 = Console.ReadLine().Split(' ');

name3 = input3[0];

code3 = uint.Parse(input3[1]);

time1\_sgi1 = float.Parse(input3[2]);

time1\_sgi2 = float.Parse(input3[3]);

// Налаштування кольорів для різних частин таблиці

Console.ForegroundColor = ConsoleColor.Green; // Колір для рамки таблиці

Console.WriteLine("┌───────────────────┬───────────────────────┬────────────────────────┬──────────────────┐");

Console.WriteLine("│ Час виконання деяких програм, які застосовують паралельні алгоритми │");

Console.WriteLine("├───────────────────┬───────────────────────┬────────────────────────┬──────────────────┤");

Console.WriteLine("│ Назва програми │ Кількість рядків коду │ Час SGI Challenge (сек) │ Час SGI Indy (сек)│");

Console.WriteLine("├───────────────────┼───────────────────────┼────────────────────────┼──────────────────┤");

// Виведення даних для кожної програми з різними кольорами

Console.ForegroundColor = ConsoleColor.Yellow; // Колір для тексту (назви програм)

Console.Write("│ {0,-17} │", name1);

Console.ForegroundColor = ConsoleColor.Cyan; // Колір для чисел (кількість рядків і час)

Console.Write(" {0,21} │ {1,22:F2} │ {2,16:F2} │\n", code1, time1\_sgi1, time1\_sgi2);

Console.ForegroundColor = ConsoleColor.Yellow;

Console.Write("│ {0,-17} │", name2);

Console.ForegroundColor = ConsoleColor.Cyan;

Console.Write(" {0,21} │ {1,22:F2} │ {2,16:F2} │\n", code2, time1\_sgi1, time1\_sgi2);

Console.ForegroundColor = ConsoleColor.Yellow;

Console.Write("│ {0,-17} │", name3);

Console.ForegroundColor = ConsoleColor.Cyan;

Console.Write(" {0,21} │ {1,22:F2} │ {2,16:F2} │\n", code3, time1\_sgi1, time1\_sgi2);

// Виведення нижньої межі таблиці

Console.ForegroundColor = ConsoleColor.Green;

Console.WriteLine("└───────────────────┴───────────────────────┴────────────────────────┴──────────────────┘");

// Повернення кольору до стандартного для інших виводів

Console.ResetColor();

}

}  
  
﻿/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* Лабораторна робота №3 \*/

/\* Арифметичні операції та математичні функції \*/

/\* мови C# \*/

/\* Приклад вирішення. Варіант №14. \*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

using System;

class Program

{

static void Main()

{

// Вхідні дані

Console.Write("Введіть значення змінних n, x, a: ");

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

double x = double.Parse(Console.ReadLine());

double a = double.Parse(Console.ReadLine());

// Обчислення t1

double t1 = (1 / (2 \* (n - 1) \* Math.Pow(x, n - 1))) + (a / (2 \* n \* Math.Pow(x, n)));

// Обчислення t2

double t2 = (1 / (2 \* a)) \* Math.Pow(Math.Tan(a \* x), 2) + (1 / a) \* Math.Log(Math.Cos(a \* x));

// Виведення результатів

Console.WriteLine($"\nt1 = {t1:F6}");

Console.WriteLine($"t2 = {t2:F6}");

}

}  
  
﻿using System;

class Program

{

static void Main()

{

try

{

// Вхідні дані

Console.Write("Введіть значення змінних n, x, a: ");

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

double x = double.Parse(Console.ReadLine());

double a = double.Parse(Console.ReadLine());

// Перевірка на ділення на нуль і недопустимі значення

if (n == 1)

{

throw new ArgumentException("Значення n не може бути рівним 1, оскільки це призведе до ділення на нуль.");

}

if (a == 0)

{

throw new ArgumentException("Значення a не може бути рівним 0, оскільки це призведе до ділення на нуль.");

}

if (Math.Cos(a \* x) <= 0)

{

throw new ArgumentException("Косинус не може бути меншим або рівним 0 для логарифма, оскільки це призведе до математичної помилки.");

}

// Обчислення t1

double t1 = (1 / (2 \* (n - 1) \* Math.Pow(x, n - 1))) + (a / (2 \* n \* Math.Pow(x, n)));

// Обчислення t2

double t2 = (1 / (2 \* a)) \* Math.Pow(Math.Tan(a \* x), 2) + (1 / a) \* Math.Log(Math.Cos(a \* x));

// Виведення результатів

Console.WriteLine($"\nt1 = {t1:F6}");

Console.WriteLine($"t2 = {t2:F6}");

}

catch (FormatException)

{

Console.WriteLine("Помилка: введені некоректні дані. Будь ласка, переконайтеся, що вводите числові значення.");

}

catch (ArgumentException ex)

{

Console.WriteLine($"Помилка: {ex.Message}");

}

catch (DivideByZeroException)

{

Console.WriteLine("Помилка: спроба ділення на нуль.");

}

catch (Exception ex)

{

Console.WriteLine($"Невідома помилка: {ex.Message}");

}

finally

{

Console.WriteLine("Обчислення завершено.");

}

}

}  
  
﻿using System;

class Program

{

static void Main()

{

// Введення координат точки

Console.Write("Введіть координату x: ");

double x = double.Parse(Console.ReadLine());

Console.Write("Введіть координату y: ");

double y = double.Parse(Console.ReadLine());

// Перевірка, чи точка потрапляє у прямокутник

bool isInRectangle = (x >= 0 && x <= 1 && y >= 0 && y <= 1);

// Перевірка, чи точка потрапляє у півколо

bool isInSemicircle = (x \* x + y \* y <= 1 && y <= 0);

// Якщо точка належить або до прямокутника, або до півкола

if (isInRectangle || isInSemicircle)

{

Console.WriteLine("Точка потрапляє в задану область.");

}

else

{

Console.WriteLine("Точка не потрапляє в задану область.");

}

}

}  
  
﻿using System;

class Program

{

static void Main()

{

// Введення координат точки

Console.Write("Введіть координату x: ");

double x = double.Parse(Console.ReadLine());

Console.Write("Введіть координату y: ");

double y = double.Parse(Console.ReadLine());

// Перевірка, чи точка потрапляє у прямокутник

bool isInRectangle = (x >= 0 && x <= 1 && y >= 0 && y <= 1);

// Перевірка, чи точка потрапляє у півколо

bool isInSemicircle = (x \* x + y \* y <= 1 && y <= 0);

// Виведення результату за допомогою тернарного оператора

Console.WriteLine(isInRectangle || isInSemicircle

? "Точка потрапляє в задану область."

: "Точка не потрапляє в задану область.");

}

}  
  
﻿using System;

class Program

{

static void Main()

{

// Задана точність

const double eps = 0.000001;

double sum = 0; // Сума членів ряду

double term; // Поточний член ряду

long n = 0; // Номер члена ряду

// Основний цикл

while (true)

{

// Обчислення чергового члена ряду

term = Math.Pow(-1, n) \* (n + 1) / (Math.Pow(n, 3) + 1);

// Перевірка досягнення межі точності

if (Math.Abs(term) >= eps)

{

sum += term; // Накопичення суми

}

else

{

break; // Вихід з циклу, якщо досягнуто точності

}

// Якщо враховано 10 членів ряду, виведення суми

if (n == 9)

{

Console.WriteLine($"Сума 10-ти членів ряду = {sum:F7}");

}

n++; // Переходимо до наступного члена ряду

}

// Виведення остаточного значення суми

Console.WriteLine($"Повна сума ряду = {sum:F7}");

}

}  
  
﻿using System;

class Program

{

static void Main()

{

// Введення похибки та кількості членів ряду

Console.Write("Введіть точність (eps): ");

double eps = double.Parse(Console.ReadLine());

Console.Write("Введіть максимальну кількість членів ряду: ");

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

double sum = 0; // Сума членів ряду

double term; // Поточний член ряду

int n = 0; // Лічильник членів ряду

// Основний цикл для обчислення суми ряду

while (n < maxTerms)

{

// Обчислення чергового члена ряду

term = Math.Pow(-1, n) \* (n + 1) / (Math.Pow(n, 3) + 1);

// Якщо поточний член за абсолютним значенням менший за похибку, виходимо з циклу

if (Math.Abs(term) < eps)

{

Console.WriteLine($"Досягнута похибка eps = {eps} на {n + 1}-му члені ряду.");

break;

}

// Накопичення суми

sum += term;

// Якщо обчислено 10 членів ряду, вивести проміжну суму

if (n == 9)

{

Console.WriteLine($"Сума 10-ти членів ряду = {sum:F7}");

}

n++; // Переходимо до наступного члена ряду

}

// Перевірка, чи обчислено повну кількість членів ряду

if (n == maxTerms)

{

Console.WriteLine($"Досягнута максимальна кількість членів ряду = {maxTerms}.");

}

// Виведення остаточного значення суми

Console.WriteLine($"Повна сума ряду = {sum:F7}");

}

}  
  
﻿using System;

class Program

{

static void Main()

{

// Зовнішній цикл - для 5 періодів

for (int n = 0; n < 5; n++)

{

// Виведення заголовка таблиці

Console.WriteLine("| x | y |");

Console.WriteLine("|-------|------------|");

// Внутрішній цикл для одного періоду

for (double x = 0; x < 4; x += 0.25)

{

double y;

// 1-й відрізок: лінійний графік

if (x < 1)

{

y = 2 \* x - 1;

}

// 2-й відрізок: частина кола

else if (x < 3)

{

y = 1 - Math.Sqrt(1 - (x - 2) \* (x - 2));

}

// 3-й відрізок: спадна лінія

else

{

y = 7 - 2 \* x;

}

// Виведення рядка таблиці

Console.Write($"| {x + n \* 4,5:F2} | {y,10:F7} |");

// Визначення позиції точки для графіка

int h = (int)((y + 1) \* 10);

if ((y - 1) \* 10 - h > 0.5)

{

h++;

}

// Виведення точки на графіку

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

{

Console.Write(" ");

}

Console.WriteLine("\*");

}

// Пауза до введення команди оператора

Console.WriteLine("Натисніть клавішу Enter для продовження...");

Console.ReadLine();

}

}

}  
﻿using System;

class Program

{

static void Main()

{

Console.BackgroundColor = ConsoleColor.Black;

Console.ForegroundColor = ConsoleColor.White;

// Зовнішній цикл - для 5 періодів

for (int n = 0; n < 5; n++)

{

// Виведення заголовка таблиці

Console.ForegroundColor = ConsoleColor.Green;

Console.WriteLine("| x | y |");

Console.WriteLine("|-------|------------|");

Console.ForegroundColor = ConsoleColor.White;

// Внутрішній цикл для одного періоду

for (double x = 0; x < 4; x += 0.25)

{

double y;

// 1-й відрізок: лінійний графік

if (x < 1)

{

y = 2 \* x - 1;

}

// 2-й відрізок: частина кола

else if (x < 3)

{

y = 1 - Math.Sqrt(1 - (x - 2) \* (x - 2));

}

// 3-й відрізок: спадна лінія

else

{

y = 7 - 2 \* x;

}

// Виведення рядка таблиці

Console.Write($"| {x + n \* 4,5:F2} | {y,10:F7} |");

// Визначення позиції точки для графіка

int h = (int)((y + 1) \* 10);

if ((y - 1) \* 10 - h > 0.5)

{

h++;

}

// Виведення точки на графіку

Console.ForegroundColor = ConsoleColor.Cyan;

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

{

Console.Write(" ");

}

Console.Write("⸜"); // Використовуємо символ ⸜ для точки на графіку

Console.WriteLine();

Console.ForegroundColor = ConsoleColor.White;

}

// Пауза до введення команди оператора

Console.ForegroundColor = ConsoleColor.Yellow;

Console.WriteLine("Натисніть клавішу Enter для продовження...");

Console.ReadLine();

Console.ForegroundColor = ConsoleColor.White;

}

}

}