

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-22.05- 05.02/2/125.00.1.Б/ОК14- 2024
	Екземпляр № 1	Арк __ / 6

Лабораторна робота №2. Вступ до платформи Microsoft .NET та основ програмування на C#

Мета роботи: навчитись застосовувати оператори керування потоком виконання програми (if, switch, for, while, do ... while) у мові C# для розробки алгоритмів.

Теоретичні відомості

Введення та виведення інформації в консольному проєкті є базовим елементом взаємодії програми з користувачем. Для цього у C# використовується клас Console, який забезпечує різноманітні методи для роботи з текстом у консолі. Для виведення даних у консоль використовуються методи Console.Write() та Console.WriteLine(). Метод Write() виводить текст без перенесення курсора на новий рядок, а WriteLine() – автоматично додає перенесення. Це дозволяє організувати зручне форматування виводу. Наприклад

```
Console.Write("Це текст на тому ж рядку.");  
Console.WriteLine("Це текст на новому рядку.");
```

Метод Console.WriteLine() підтримує форматування рядків, що дозволяє виводити змінні у визначеному форматі. Для цього використовуються фігурні дужки {}:

```
int age = 25;  
string name = "Анна";  
Console.WriteLine("Привіт, {0}! Тобі {1} років.", name, age);
```

Або сучасний спосіб із використанням інтерполяції рядків:

```
Console.WriteLine($"Привіт, {name}! Тобі {age} років.");
```

Метод Console.ReadLine() використовується для отримання тексту, введеного користувачем. Результат завжди повертається у вигляді рядка (тип string). Для роботи з іншими типами даних необхідне перетворення:

```
Console.Write("Введіть своє ім'я: ");  
string name = Console.ReadLine();  
  
Console.Write("Введіть свій вік: ");  
int age = int.Parse(Console.ReadLine());  
Console.WriteLine($"Вітаю, {name}. Вам {age} років.");
```

Оскільки метод Console.ReadLine() завжди повертає рядок, варто передбачити обробку можливих помилок, наприклад, при конвертації:

```
Console.Write("Введіть число: ");  
string input = Console.ReadLine();  
if (int.TryParse(input, out int number))  
{  
    Console.WriteLine($"Ви ввели число: {number}");  
}  
else  
{  
    Console.WriteLine("Помилка: введене значення не є числом.");  
}
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-22.05- 05.02/2/125.00.1.Б/ОК14- 2024
	Екземпляр № 1	Арк __/7

Для покращення зручності взаємодії з користувачем можна змінювати кольори тексту та фону:

```
Console.ForegroundColor = ConsoleColor.Green;
Console.WriteLine("Це текст зеленого кольору.");
Console.ResetColor(); // Повертає стандартні налаштування кольору
```

Типи даних у C# та способи їх парсингу

Тип даних	Опис	Метод парсингу	Приклад використання
int	Цілі числа, наприклад, 1, 42, -100.	int.Parse()	int number = int.Parse(Console.ReadLine());
		int.TryParse()	if (int.TryParse(Console.ReadLine(), out int number)) { /* успішно */ }
double	Дійсні числа з плаваючою крапкою, наприклад, 1.23, -4.56.	double.Parse()	double number = double.Parse(Console.ReadLine());
		double.TryParse()	if (double.TryParse(Console.ReadLine(), out double number)) { /* успішно */ }
float	Менш точні дійсні числа порівняно з double, займають менше пам'яті.	float.Parse()	float number = float.Parse(Console.ReadLine());
		float.TryParse()	if (float.TryParse(Console.ReadLine(), out float number)) { /* успішно */ }
decimal	Високоточні числа, зазвичай використовуються для фінансових розрахунків.	decimal.Parse()	decimal number = decimal.Parse(Console.ReadLine());
		decimal.TryParse()	if (decimal.TryParse(Console.ReadLine(), out decimal number)) { /* успішно */ }
bool	Логічний тип, що приймає значення true або false.	bool.Parse()	bool flag = bool.Parse(Console.ReadLine());
		bool.TryParse()	if (bool.TryParse(Console.ReadLine(), out bool flag)) { /* успішно */ }
char	Один символ, наприклад, 'a', '1', '?'. Пряма конвертація	char.Parse()	char letter = char.Parse(Console.ReadLine());
		Пряма конвертація	char letter = Console.ReadLine()[0];
string	Текстові рядки, наприклад, "Hello World". Введення без парсингу	Введення без парсингу	string text = Console.ReadLine();
DateTime	Дата і час, наприклад, "01.01.2025 12:30:00".	DateTime.Parse()	DateTime date = DateTime.Parse(Console.ReadLine());
		DateTime.TryParse()	if (DateTime.TryParse(Console.ReadLin

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-22.05- 05.02/2/125.00.1.Б/ОК14- 2024
	Екземпляр № 1	Арк __/8

			e(), out DateTime date)) { /* успішно */ }
long	Цілі числа великого діапазону, наприклад, 9223372036854775807.	long.Parse()	long bigNumber = long.Parse(Console.ReadLine());
		long.TryParse()	if (long.TryParse(Console.ReadLine(), out long bigNumber)) { /* успішно */ }

Умовний оператор if-else дозволяє виконувати різні блоки коду в залежності від виконання задано умови. Синтаксис:

```
if (умова)
{
    // Блок коду, що виконується, якщо логічний_вираз істинний
}
else
{
    // Блок коду, що виконується, якщо логічний_вираз хибний (необов'язковий)
}
```

Конструкція if-else if-else дозволяє перевіряти кілька умов послідовно і виконувати відповідний блок коду, коли одна з умов є істинною. Це особливо корисно, коли у вас є кілька можливих шляхів виконання програми залежно від різних ситуацій. Синтаксис:

```
if (умова1)
{
    // Код, який виконається, якщо умова1 істинна
}
else if (умова2)
{
    // Код, який виконається, якщо умова1 хибна, а умова2 істинна
}
else if (умова3)
{
    // Код, який виконається, якщо умова1 і умова2 хибні, а умова3 істинна
}
// ... Можна додати більше блоків else if за потреби
else
{
    // Код, який виконається, якщо всі попередні умови хибні (необов'язковий блок)
}
```

Оператор множинного вибору switch забезпечує ефективну реалізацію множинного розгалуження. В C# вираз у операторі switch може бути тільки цілочисельного типу, рядком, символом або перерахуванням. Синтаксис:

```
switch (вираз)
{
    case значення1:
        // Код для значення1
        break;
    case значення2:
        // Код для значення2
        break;
    // ... інші варіанти
    default:
        // Код, що виконується, якщо жоден з варіантів не підійшов (необов'язковий)
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-22.05- 05.02/2/125.00.1.Б/ОК14- 2024
	Екземпляр № 1	Арк __/9

}

Циклічні оператори дозволяють багаторазово виконувати блок коду. Цикл `for` виконує блок коду, доки вказаний логічний вираз обчислюється як `true`. Цикл `while` виконується доки задана умова залишається істинною. Цикл `do-while` схожий на `while`, але гарантовано виконує тіло циклу хоча б один раз. Цикл `foreach` використовується для ітерації по елементах колекції, масиву чи списку. У ньому не потрібно вказувати індекси, що робить його зручним для роботи з даними. Синтаксис:

```
// for
for (ініціалізація; умова_продовження; ітерація)
{
    // Тіло циклу
}

// while
while (умова)
{
    // Тіло циклу
}

// do-while
do
{
    // Тіло циклу
} while (умова);

// foreach
string[] fruits = { "Яблуко", "Банан", "Апельсин" };
foreach (string fruit in fruits)
{
    Console.WriteLine(fruit);
}
```

Для роботи з математичними операціями потрібно використовувати клас `Math`. Він надає розробнику широкий спектр статичних методів, що дозволяють здійснювати фундаментальні математичні операції, включаючи тригонометричні, логарифмічні, степеневі функції, операції округлення та інші. Завдяки цьому класу, розробник має змогу ефективно виконувати необхідні обчислення безпосередньо в коді програми, що значно спрощує процес розробки та підвищує його продуктивність. Наприклад, обчислення площі трикутника за трьома його сторонами, використовуючи формулу Герона, виглядатиме так:

```
// Формула Герона
// Площа = √(p(p - a)(p - b)(p - c))

// Реалізація
double a = 5.0;
double b = 6.0;
double c = 7.0;

// Обчислення півпериметра
double p = (a + b + c) / 2;
// Обчислення площі за формулою Герона
double area = Math.Sqrt(p * (p - a) * (p - b) * (p - c));
Console.WriteLine($"Площа трикутника дорівнює: {area}");
```

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-22.05- 05.02/2/125.00.1.Б/ОК14- 2024
	Екземпляр № 1	Арк __ / 10

Для налаштування української локалізації в консолі необхідно додати такий код:

```
using System.Text;
...
Console.OutputEncoding = Encoding.UTF8;
```

Зміст роботи

Для кожного з завдань побудувати блок-схему!

Завдання 1: Створити приватний репозиторій з назвою oop-lab2. Надати доступ викладачу.

Завдання 2: Створити консольний проект Visual C#. Назва рішення – OOPLab2. Назва проекту - OOPLab2.2

2.1 Приєднати віддалений репозиторій до локального

2.2 Реалізувати обчислення за заданою формулою згідно обраного варіанту, x, y, z вводить користувач

1	$s = \frac{3 \sin(x^2) + \cos(y^3)}{e^z + \frac{1}{1+z^2}}$
2	$s = \sqrt[5]{x^2 - y^2 + 1} \tan\left(z + \frac{\pi}{4}\right) - \frac{z}{ z + 1}$
3	$s = \frac{e^{x+y} - \sin(z)}{x^2 + y^2 + z^2 + 1} + \frac{x}{ x + y }$
4	$s = (1 + \cos(x))^{\sin(y)} - \frac{y}{1 + x^3}$
5	$s = \sqrt[3]{x^3 + y^3 + z^3} \cos(z) + x - y $
6	$s = \frac{2 \cos(x^2) - \frac{1}{2}}{\frac{1}{2} + \sin(y^{2-z})}$
7	$s = \frac{-x + \sqrt{x^2 - 4yz}}{2y}$
8	$s = \frac{\sqrt[4]{y + \sqrt[3]{x-1}}}{ x - y (\sin(z^2) + \tan(z))}$

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-22.05- 05.02/2/125.00.1.Б/ОК14- 2024
	Екземпляр № 1	Арк __/11

9	$s = 2^{y^x} + (3^x)^y - \frac{y(\tan(z) - \frac{1}{3})}{ x + \frac{1}{y^2 + 1}}$
10	$s = 5 \cos(x) - \frac{z}{4} \sin(y) \frac{4 z - x - 5}{z(x - y) + z^2}$
11	$s = \frac{\sqrt[3]{3 + (x - y)}}{x^2 + z^3 + 4} - \tan(z)$
12	$s = \frac{e^{z+y}(z - y)^{x+z}}{\sin(x) + \sin(y)} + \sqrt[4]{x^7 + \ln(y)}$
13	$s = 2^{-x} \sqrt{x + \sqrt[4]{ y ^3} \sqrt[3]{e^{x-1}/\sin(z)}}$
14	$s = 2^{y^4} + \cos(x^2) \frac{ x - y \left(\frac{(\sin(z))^2}{e^{ x-y } + \frac{x}{2}} \right)}{\sqrt{x + y}}$
15	$s = \frac{x^{y+y} e^{y-1}}{1 + x y - \tan(z) } (1 + y - x) + \frac{ y - x ^2}{2} - \frac{ y - x ^3}{3}$

2.3 Створити коміт та завантажити зміни в віддалений репозиторій

Завдання 3: Додати новий проект з назвою OOPLab2.3 в існуюче рішення. Числа повинен вводити користувач.

3.1 Знайти косинус мінімального з 4 заданих чисел

3.2 По даному числу (1-12) потрібно вивести назву відповідного йому місяця.

3.3 Вивести на екран синус максимального з 3 заданих чисел

3.4 Написати програму обчислення вартості покупки з урахуванням знижки. Знижка в 5% надається в тому випадку, якщо сума покупки більше 1000 грн, в 10% - якщо сума більше 2000 грн.

3.5 Створити коміт та завантажити зміни в віддалений репозиторій

Завдання 4: Додати новий проект з назвою OOPLab2.4 в існуюче рішення.

4.1 Написати програму, яка перевіряє чи є введене користувачем число простим (використовувати цикл for)

4.2 Написати програму для обчислення факторіала числа N. Число N вводить користувач. Реалізувати за допомогою циклу for та циклу while.

4.3 Дано натуральне число n. Підрахувати:

Житомирська політехніка	МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА» Система управління якістю відповідає ДСТУ ISO 9001:2015	Ф-22.05- 05.02/2/125.00.1.Б/ОК14- 2024
	Екземпляр № 1	Арк __ / 12

1	$\frac{1}{1} + \frac{1}{2} + \dots + \frac{1}{n}$	9	$\frac{1}{1^5} + \frac{1}{2^5} + \dots + \frac{1}{n^5}$
2	$\frac{1}{3^2} + \frac{1}{5^2} + \dots + \frac{1}{(2n+1)^2}$	10	$\left(1 + \frac{1}{1^2}\right) + \left(1 + \frac{1}{2^2}\right) + \dots + \left(1 + \frac{1}{n^2}\right)$
3	$\frac{1}{\sin(1)} + \frac{1}{\sin(1) + \sin(2)} + \dots + \frac{1}{\sin(1) + \sin(n)}$	11	$(1 + \sin(0,1)) + (1 + \sin(0,2)) + \dots + 1 + \sin(5)$
4	$\frac{\sqrt{2}}{1} + \frac{\sqrt{2}}{2} + \dots + \frac{\sqrt{2}}{n}$	12	$\frac{2}{1} + \frac{3}{2} + \dots + \frac{n+1}{n}$
5	$\frac{2}{1^2} + \frac{3}{2^2} + \dots + \frac{n+1}{n^2}$	13	$1 * 2 + 2 * 3 + 3 * 4 + \dots + n(n+1)$
6	$1! + 2! + 3! + \dots + n! (n > 1)$	14	$\left(1 + \frac{1}{2 + 1^2}\right) + \left(1 + \frac{1}{3 + 2^2}\right) + \dots + \left(1 + \frac{1}{(n+1) + n^2}\right)$
7	$\left(1 + \frac{1}{1^2}\right) - \left(1 + \frac{1}{2^2}\right) - \dots - \left(1 + \frac{1}{n^2}\right)$	15	$\frac{1}{1 * 2} + \frac{1}{2 * 3} + \dots + \frac{1}{n(n+1)}$
8	$\sin(x) + \sin(x^2) + \sin(x^3) + \dots + \sin(x^n)$		

4.4 Створити коміт та завантажити зміни в віддалений репозиторій

Контрольні запитання

1. Які методи класу Console використовуються для виведення інформації в консоль?
2. Чим відрізняються методи Console.Write() та Console.WriteLine()?
3. Як можна організувати багаторядковий вивід у консоль?
4. Для чого використовується метод Console.ReadLine()?
5. Яким чином можна перевірити, чи користувач ввів коректне число?
6. Що відбувається, якщо користувач вводить некоректний тип даних для парсингу?
7. Що таке форматований вивід, і як працює метод Console.WriteLine() з параметрами (наприклад, "{0} {1}")?
8. Що таке ітерація циклу? Як вона здійснюється в різних типах циклів?
9. Які ключові слова використовуються для керування виконанням циклів? Поясніть їх призначення та наведіть приклади їх використання.
10. Які умовні оператори існують в C#? Поясніть їх призначення та синтаксис.