### Навчальна дисципліна «Об’єктно-орієнтоване програмування»

### **ЛАБОРАТОРНА РОБОТА #0101**

## Тема:

## Основи програмування мовою C#

Варіант №1

**Виконав**:  
студент 2-го курсу  
групи KN1-B23  
Мартиновський А.В

Кам’янець-Подільський – 2025

**Теоретичні відомості:**

1. **Структура програми**  
   Програма в C# складається з простору імен (namespace), в якому знаходяться класи. Головний клас містить метод Main(), що є точкою входу. Всередині Main() викликаються інші методи та виконуються команди.
2. **Змінні та константи**  
   Змінні – це області пам’яті, що можуть змінювати значення під час виконання (int, double, string, bool). Константи (const) зберігають незмінні значення. Тип змінної визначає, які дані можна в ній зберігати.
3. **Консольне введення-виведення**  
   Дані можна виводити у консоль через команди виводу (WriteLine, Write). Введення даних виконується через зчитування рядка з клавіатури, після чого його можна конвертувати у потрібний тип.
4. **Арифметичні, побітові операції, операції присвоєння, перетворення типів**  
   Арифметичні операції (+, -, \*, /, %) використовуються для обчислень. Побітові (&, |, ^, <<, >>) працюють із бітами чисел. Операції присвоєння (=, +=, -=, \*=, /=) змінюють значення змінних. Перетворення типів бувають неявними (наприклад, int у double) і явними (з використанням операторів або спеціальних методів).
5. **Умовні вирази**  
   Перевірка умов здійснюється через if-else, що дозволяє виконувати різні команди залежно від умови. Тернарний оператор (?:) є скороченою формою if-else для простих умовних виразів.
6. **Цикли**  
   Цикли використовуються для багаторазового виконання коду. for застосовується, коли кількість ітерацій відома. while виконується, поки виконується умова. do-while виконує код хоча б один раз. foreach використовується для обходу колекцій і масивів.
7. **Масиви**  
   Масив – це структура, що містить набір значень одного типу. Індексація починається з нуля. Масиви можуть бути одновимірними, двовимірними та багатовимірними. Для обробки масивів часто використовуються цикли.
8. **Методи, передача параметрів, return**  
   Методи дозволяють організувати код у повторно використовувані блоки. Вони можуть повертати значення (return). Параметри передаються за значенням (копія) або за посиланням (ref, out). params дозволяє передавати змінну кількість аргументів.
9. **Рекурсія та локальні функції**  
   Рекурсія – це коли метод викликає сам себе для вирішення задачі. Використовується для факторіалу, пошуку в структурах даних тощо. Локальні функції визначаються всередині іншого методу й доступні тільки в його межах.
10. **Конструкція switch**  
    Використовується для перевірки одного значення на відповідність кільком можливим варіантам (case). default визначає поведінку у випадку, якщо жоден case не спрацював.
11. **Перерахування (enum)**  
    Перерахування дозволяють створювати набір іменованих констант. Це корисно для роботи з категоріями, днями тижня, станами процесів тощо.

**Завдання до виконання:**

Задача 1. Сума двох чисел. Програма повинна зчитувати два цілі числа з клавіатури, знаходити їх суму та виводити результат.

Задача 2. Куб числа. Розробити програму, яка обчислює куб введеного числа. Задача 3. Максимум із трьох. Написати програму, яка зчитує три числа та визначає найбільше з них.

**Послідовність виконання завдання:**

**Завдання 1: Сума двох чисел (C#)**

1. Відкрити середовище розробки (в моєму випадку я використав онлайн компілятор GDB Online Complier).
2. Підключаємо стандартну бібліотеку в C#:

using System;

1. Створюємо клас:

public class Sum

1. Підключаємо UTF-8 для того щоб компілятор розпізнавав українську мову:

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

1. Пишемо блоки вводу числа в програму:

Console.Write("Введіть перше число: ");

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

Console.Write("Введіть друге число: ");

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

1. Обчислюємо суму двох чисел:

int sum = a + b;

1. Виводимо фінальне значення:

Console.WriteLine($"Сума: {sum}");

1. Запускаємо програму.
2. Вводимо два числа у терміналі та перевіряємо результат.

**Завдання 2: Куб числа (C#)**

1. Відкрити середовище розробки (в моєму випадку я використав онлайн компілятор GDB Online Complier).
2. Підключаємо стандартну бібліотеку в C#:

using System;

1. Створюємо клас:

public class Pow

1. Підключаємо UTF-8 для того щоб компілятор розпізнавав українську мову:

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

1. Пишемо блок вводу числа в програму:

Console.Write("Введіть число: ");

double a = int.Parse(Console.ReadLine());Обчислюємо суму двох чисел:

int sum = a + b;

1. Обчислюємо куб числа:

double b = Math.Pow(a, 3);

1. Виводимо фінальне значення:

Console.WriteLine($"Результат: {b}");

1. Запускаємо програму.
2. Вводимо число у терміналі та перевіряємо результат.

**Завдання 3: Максимум із трьох (C#)**

1. Відкрити середовище розробки (в моєму випадку я використав онлайн компілятор GDB Online Complier).
2. Підключаємо стандартну бібліотеку в C#:

using System;

1. Створюємо клас:

public class Max

1. Підключаємо UTF-8 для того щоб компілятор розпізнавав українську мову:

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

1. Пишемо блоки вводу числа в програму:

Console.Write("Введіть перше число: ");

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

Console.Write("Введіть друге число: ");

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

Console.Write("Введіть третє число: ");

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

1. Вводимо перевірку числа а:

if (a>b && a>c)

{

Console.WriteLine($"Найбільше число: {a}");

}

1. Вводимо перевірку числа b:

else if (b>a && b>c)

{

Console.WriteLine($"Найбільше число: {b}");

}

1. Якщо минулі дві перевірки хибні то виводимо c:

else

{

Console.WriteLine($"Найбільше число: {c}");

}

1. Запускаємо програму.
2. Вводимо числа у терміналі та перевіряємо результат.

**Лістинг програм:**

Завдання 1:

using System;

public class Sum

{

static void Main()

{

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

Console.Write("Введіть перше число: ");

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

Console.Write("Введіть друге число: ");

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

int sum = a + b;

Console.WriteLine($"Сума: {sum}");

}

}

Завдання 2:

using System;

public class Pow

{

static void Main()

{

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

Console.Write("Введіть число: ");

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

double b = Math.Pow(a, 3);

Console.WriteLine($"Результат: {b}");

}

}

Завдання 3:

using System;

public class Max

{

static void Main()

{

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

Console.Write("Введіть перше число: ");

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

Console.Write("Введіть друге число: ");

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

Console.Write("Введіть третє число: ");

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

if (a>b && a>c)

{

Console.WriteLine($"Найбільше число: {a}");

}

else if (b>a && b>c)

{

Console.WriteLine($"Найбільше число: {b}");

}

else

{

Console.WriteLine($"Найбільше число: {c}");

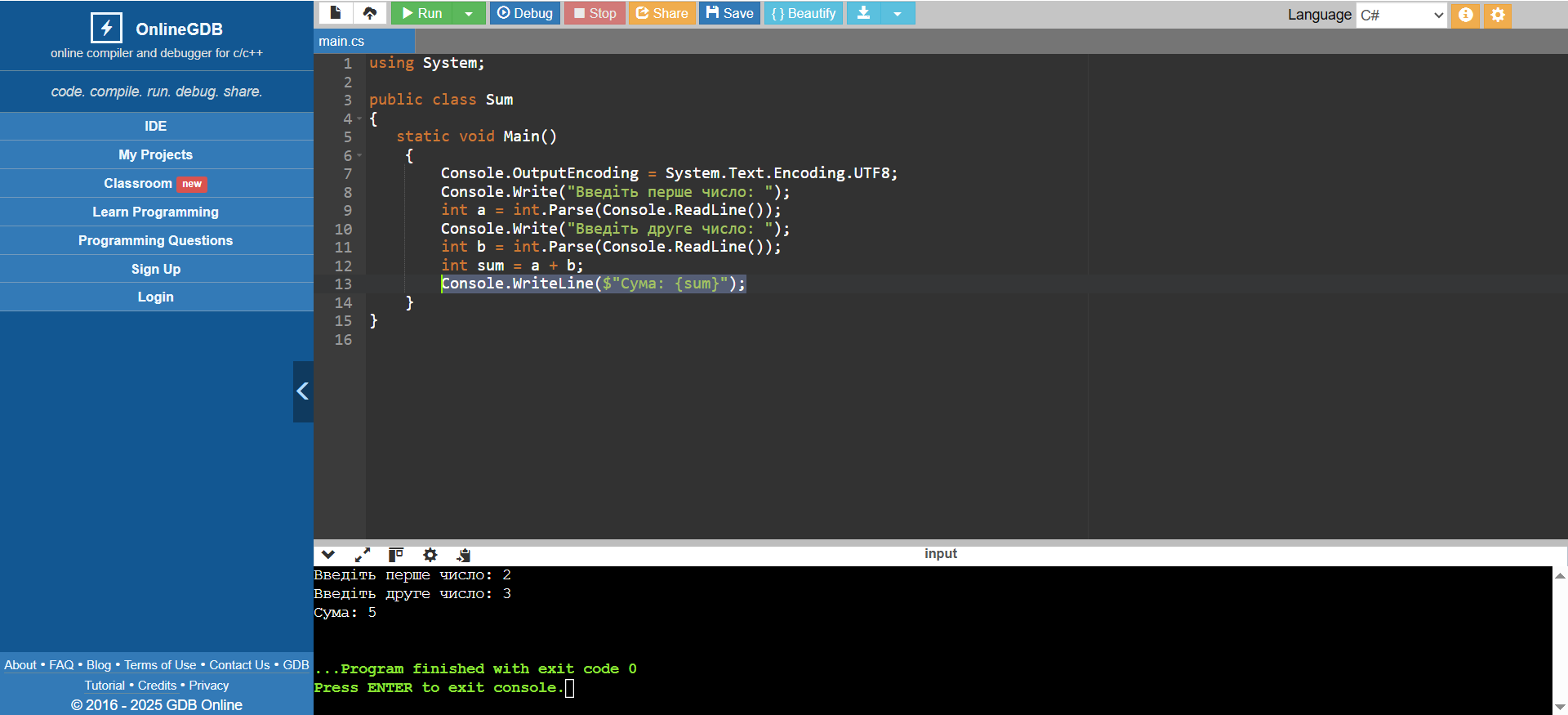
}

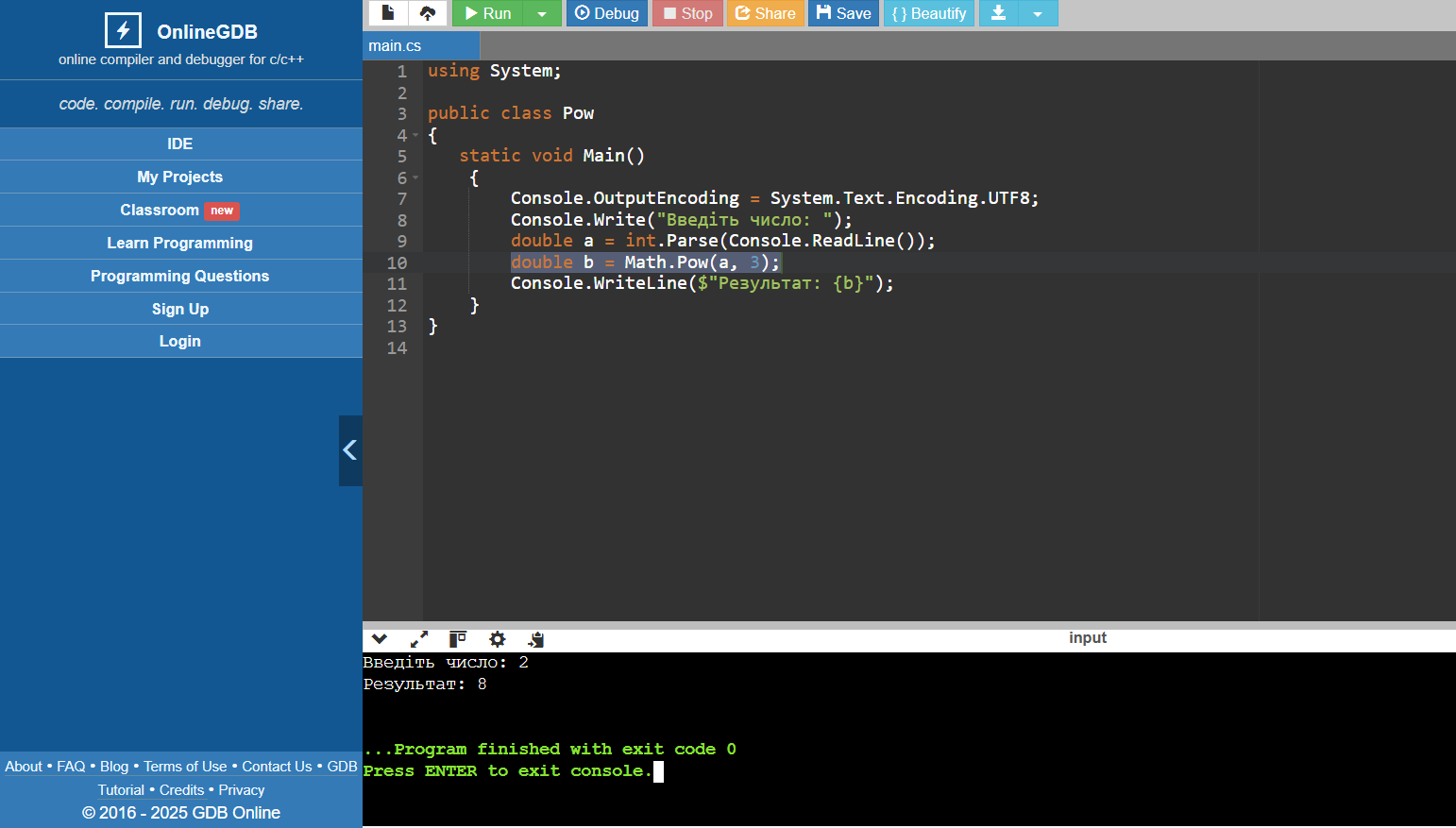
}

}

**Результати виконання програм:**

Завдання 1:



Завдання 2: 

Завдання 3: 