

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
Криворізький національний університет  
Кафедра моделювання та програмного забезпечення

Лабораторна робота №1  
Тема: «Загальномовне середовище виконання - Common Language Runtime (CLR).  
Створення  
консольного додатка CLR. Специфіка C++/CLI: введення та форматування виведення.»

Виконав студент  
групи \_\_\_\_\_

\_\_\_\_\_

ІПЗ-23-1

Курілов Артур

Перевірив викладач

\_\_\_\_\_

Білашенко Світлана Вікторівна  
Карабут Надія Олександрівна

Кривий Ріг  
202\_

### 1.1 Мета роботи

Отримати знання і навички необхідні для програмування в загальному середовищі

виконання (CLR) та навчитися використовувати їх на практиці.

### 1.2 Основні теоретичні відомості та приклади програмної реалізації.

Common Language Runtime, скорочено CLR — «загальному виконуюче середовище» — це

компонент пакету Microsoft .NET Framework, віртуальна машина, на якій виконуються всі мови

платформи .NET Framework. CLR транслює початковий код в байт-код мовою IL, реалізація компіляції

якого компанією Microsoft називається MSIL, а також надає MSIL-програмам (а отже, і програмам,

написаним на мовах високого рівня, що підтримують .NET Framework) доступ до бібліотеки класів .NET

Framework, або так званою .NET FCL (англ. Framework Class Library). Середовище CLR є реалізацією

специфікації CLI (англ. Common Language Infrastructure), специфікації загальному інфраструктури,

компанією Microsoft.

Завдання до роботи

1. Ознайомитися з основними теоретичними відомостями за темою роботи.

2. Створити новий консольний додаток CLR. Розв'язати завдання згідно отриманого варіанту.

3. Оформити звіт з роботи.

## Завдання лабораторної роботи №1

### P-1

10. Заданий рядок символів. Підрахувати, скільки разів серед символів рядка зустрічається літера X.

```
#include "stdafx.h"
#include "pch.h"
using namespace System;
using namespace System::Text;

int main(array<System::String^>^ args)
{
    // Встановлюємо кодування UTF-8 для консолі
    Console::OutputEncoding = Encoding::UTF8;

    // Привітання
    Console::WriteLine(L"Доброго дня!");

    // Заданий рядок символів
    String^ input;

    // Запитуємо рядок у користувача
    Console::Write(L"Введіть рядок символів: ");
    input = Console::ReadLine();

    // Лічильник для літери 'X' (латинська та кирилична)
    int count = 0;

    // Проходимо по кожному символу у рядку та рахуємо кількість 'X' та 'X'
    for (int i = 0; i < input->Length; i++)
    {
```

```

        // Перевіряємо наявність кириличної 'X' (U+0425) або латинської 'X'
        (U+0058)
        if (input[i] == L'X' || input[i] == L'X')
        {
            count++;
        }
    }

    // Форматоване виведення результатів
    Console.WriteLine(L"У рядку: \"{0}\" літера 'X' або 'X' зустрічається {1}
разів.", input, count);

    return 0;
}

```

```

Доброго дня!
Введіть рядок символів: 123X45X
У рядку: "123X45X" літера 'X' або 'X' зустрічається 2 разів.

Доброго дня!
Введіть рядок символів: ТАЩXXXЧІФ123
У рядку: "ТАЩXXXЧ?Ф123" літера 'X' або 'X' зустрічається 3 разів.

```

## P-2

10. Дана строка S. Побудувати строки S<sub>1</sub> і S<sub>2</sub> із символів строки S, стоящих на четных и нечетных местах соответственно.

```

#include "pch.h"
using namespace System;
using namespace System::Text;

int main(array<System::String^>^ args)
{
    // Встановлюємо кодування UTF-8 для консолі
    Console::OutputEncoding = Encoding::UTF8;

    // Привітання
    Console::WriteLine(L"Доброго дня!");

    // Заданий рядок символів
    String^ input;

    // Запитуємо рядок у користувача
    Console::Write(L"Введіть рядок S: ");
    input = Console::ReadLine();

    // Рядки для символів на парних та непарних позиціях
    String^ S1 = "";
    String^ S2 = "";

    // Проходимо по кожному символу у рядку та розподіляємо по S1 (парні) та S2
    (непарні)
    for (int i = 0; i < input->Length; i++)
    {
        if (i % 2 == 0)
        {
            S1 += input[i]; // Парні індекси (i = 0, 2, 4, ...)
        }
        else
        {
            S2 += input[i]; // Непарні індекси (i = 1, 3, 5, ...)
        }
    }
}

```

```
// Виведення результатів
Console::WriteLine(L"Рядок S1 (парні символи): {0}", S1);
Console::WriteLine(L"Рядок S2 (непарні символи): {0}", S2);

return 0;
}
```

```
Доброго дня!
Введіть рядок S: 123456789
Рядок S1 (парні символи): 13579
Рядок S2 (непарні символи): 2468
```

```
Доброго дня!
Введіть рядок S: 41236789
Рядок S1 (парні символи): 4268
Рядок S2 (непарні символи): 1379
```

### Висновок

У цьому проєкті ми створили консольний CLR додаток у Microsoft Visual Studio за допомогою C++/CLI, використовуючи шаблони та інструменти, надані середовищем розробки. Основні етапи та результати проєкту включають:

1. Створення проєкту:  
Проект був створений як консольний CLR додаток, використовуючи шаблон "Console App" або "CLR Console Application". Це дозволяє використовувати можливості Common Language Runtime (CLR) у C++.
2. Автоматично згенеровані файли:  
Проект автоматично створює файл, який починається з `#include "pch.h"`. Цей файл є частиною механізму Precompiled Headers (PCH), який використовується для зменшення часу компіляції, завантажуючи часто використовувані заголовки один раз.
3. Обробка рядків:  
Реалізували кілька завдань, таких як підрахунок кількості літер 'X' у рядку і побудова нових рядків з символів, розташованих на парних і непарних позиціях. Було виявлено, що у разі включення пробілів у рядок обробка символів може бути некоректною, тому було додано умову для ігнорування пробілів.

Отже, проєкт демонструє ефективне використання можливостей CLR у C++/CLI для створення консольних додатків, обробки рядків і налаштування середовища розробки для досягнення точності та ефективності.