**Кафедра інженерії програмного забезпечення та кібербезпеки**

ЛАБОРАТОРНА РОБОТА №6

**ПЕРЕРАХУВАННЯ**

|  |  |
| --- | --- |
| **Виконав:** | *Лупина Ігор Ігорович* |
| **Група:** | *1-6* |
| **Варіант:** | *12* |
|  |  |
| **Викладач:** | *Макоєдова Валентина Олександрівна* |
| **Бали:** |  |

|  |  |
| --- | --- |
| **освітній ступінь** | **бакалавр /** bachelor |
| **галузь знань** | **12 Інформаційні технології /** Information Technology |
| **спеціальність** | **121Інженерія програмного забезпечення /** Software engineering |
|  |  |

**Київ 2023**

**Тема:** Перерахування

**Мета:** Закріпити вміння використовувати перерахування при написанні програм.

**Завдання:** Виконати задачі згідно варіанту 2

**Хід роботи**

**Задача 1:** Створити перерахування з прапорцем, у якому зберігаються назви 6 програмних продуктів - 3-ох систем для автоматизації обліку і 3-х систем для управління логістикою. Вивести перелік систем, застосовуючи константи (номер – назва). Дати можливість користувачеві ввести номери програмних продуктів, які він хотів би придбати, у вигляді списку, відокремлюючи номери комами. Вивести прізвище автора програми, переліки програмних продуктів окремо по кожній групі (облік/логістика).

Для виконання задачі 1 було написано наступну програму:

namespace lab6;

internal class Program

{

static void Main()

{

Console.WriteLine("Lupyna FIT 1-6");

Enum.GetNames<Products>()

.Select((x, i) => string.Format("{0} - {1}", i + 1, x))

.ToList()

.ForEach(Console.WriteLine);

Console.Write("Order: ");

var str = Console.ReadLine();

if (string.IsNullOrWhiteSpace(str))

{

Console.WriteLine("Invalid input");

return;

}

var flags = str

.Split(',', StringSplitOptions.TrimEntries | StringSplitOptions.RemoveEmptyEntries)

.Select(x => new { Parsed = int.TryParse(x, out var val), Value = val })

.Where(x => x.Parsed)

.Select(x => (int)Math.Pow(2, x.Value - 1))

.Where(x => Enum.IsDefined(typeof(Products), x))

.Cast<Products>();

if (!flags.Any())

{

Console.WriteLine("Invalid input");

return;

}

var groups = flags

.Select(x => new { Value = x, Type = x >= Products.LogisticsSystem1 ? "Logistics systems" : "Automation systems" })

.GroupBy(x => x.Type);

foreach (var group in groups)

{

var val = group.Select(x => x.Value).Aggregate((x, y) => x | y);

Console.WriteLine($"{group.Key}: {val}");

}

}

}

[Flags]

enum Products

{

AutomationSystem1 = 0b000001,

AutomationSystem2 = 0b000010,

AutomationSystem3 = 0b000100,

LogisticsSystem1 = 0b001000,

LogisticsSystem2 = 0b010000,

LogisticsSystem3 = 0b100000,

}

Результати виконання програми зображено на рисунках 1 – 2.

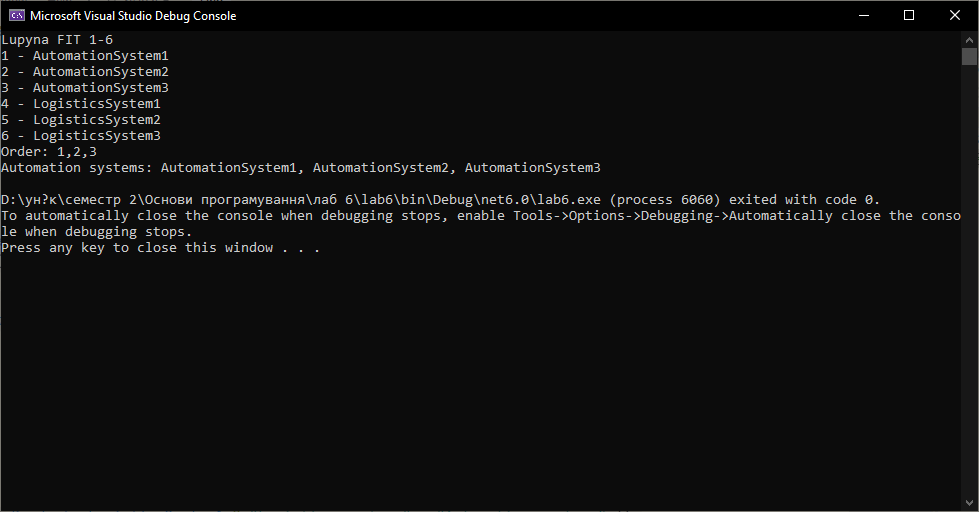


Рисунок 1 – Результат виконання програми задачі 1 з параметром 1, 2, 3

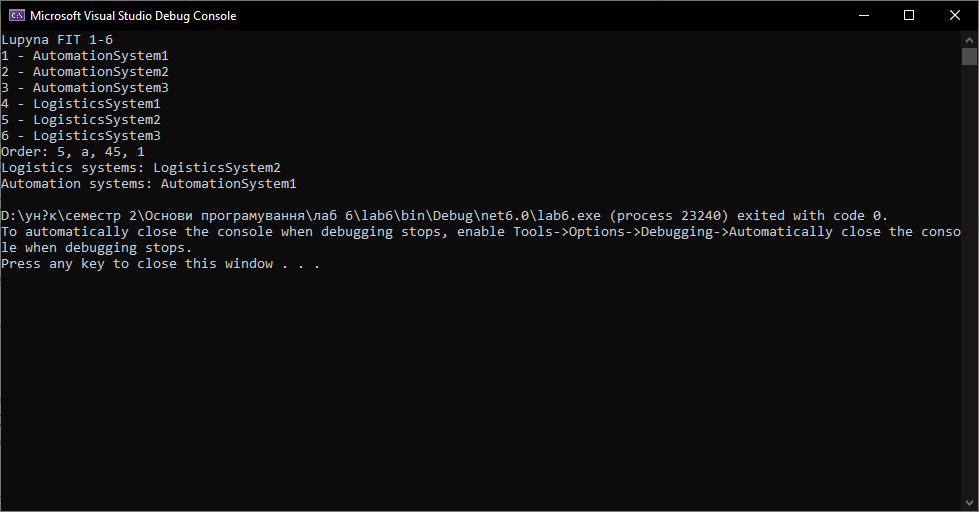


Рисунок 2 – Результат виконання програми задачі 1 з параметром 5, a, 45, 1

**Висновок:** Під час виконання практичної роботи я закріпив вміння використання перерахуваннь при написанні програм.