

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЛЬВІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ІВАНА ФРАНКА

Звіт

про виконання лабораторної роботи №6

на тему:

«Collections в C#»

Викона(в/ла):

Студент(ка) групи Фел-12

Шита М.О.

Перевірив:

Щербак С.С

Львів 2020

Мета роботи: вивчити поняття колекції, навчитися їх застосовувати.

Обладнання: ноутбук, інтегроване середовище розробки програмного забезпечення Microsoft Visual Studio (2019).

Теоретичні відомості

Колекції призначені для стандартизованої обробки груп об'єктів у програмі. Під стандартизованою обробкою розуміється використання таких відомих динамічних структур даних як: стеки, черги, лінійні списки, хеш-таблиці. Крім того, в колекціях забезпечуються поширені операції обробки масивів даних, наприклад, сортування.

У середовищі .NET Framework 4 є 5 видів колекцій:

неузагальнені;

спеціальні;

колекції з порозрядною (бітовою) організацією;

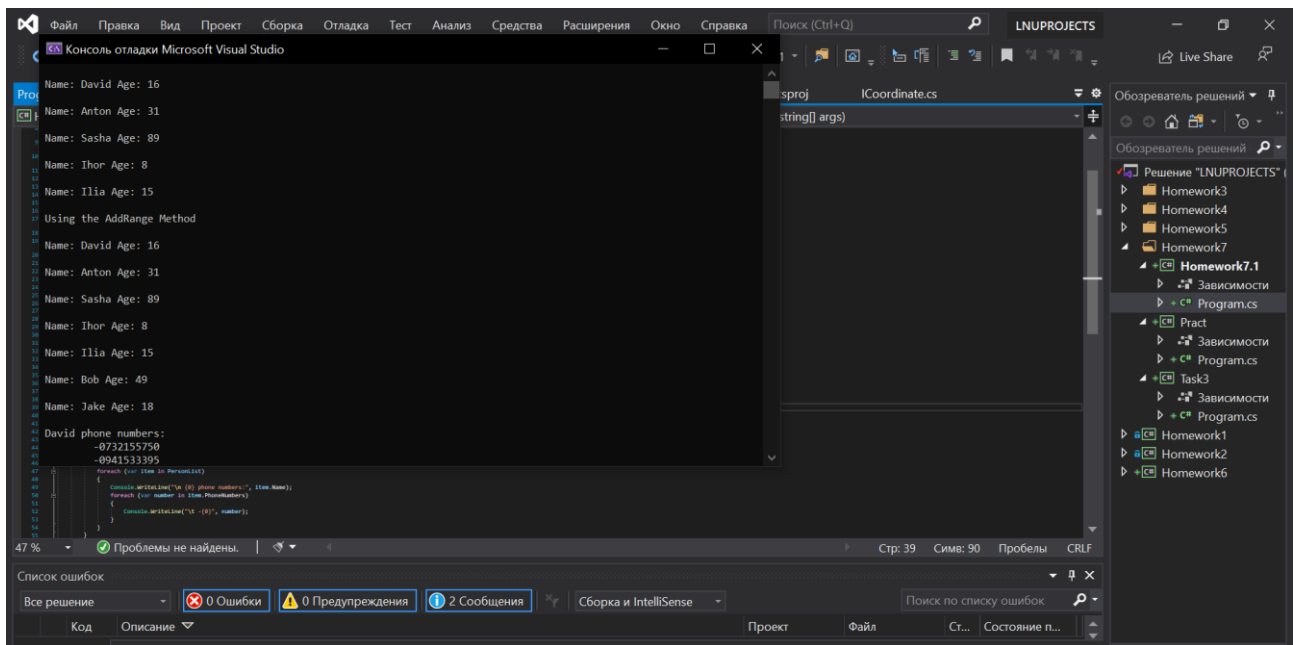
узагальнені;

паралельні (багатопотокові).

Хід роботи

1. Створити ліст в якому знаходяться кілька персон (> 5). В кожній персоні повинно бути кілька номерів (> 2). Вивести ім'я та вік кожної створеної персоні на консоль.
2. Використовуючи метод `AddRange` додати до створеного ліста ще дві персон. Вивести на консоль номери всіх персон (не використовуючи LINQ).
3. Створити список з випадково згенерованими елементами типу `string` ($n > 100$, довжина стрічки 4, всі символи великі букви). Вилучити з нього всі елементи які повторюються і які починаються з символу 'Z', відсортувати у порядку спадання (z-a). Відобразити на екрані кількість елементів в списку до і після проведених трансформацій. Створити метод `DisplayPage(int pageNumber)` при виклику якого на консоль буде виводитись відповідна сторінка оновленого списку (кількість елементів на сторінці константна: 5). Реалізувати програму так щоб номер сторінки, яку потрібно вивести, зчитувався з консолі. Якщо введена стрічка не число то завершити програму (не використовуючи LINQ).

Вивела вік, ім'я та номери на консоль.



Код до завдання 1 -2:

```
using System;
using System.Collections.Generic;

namespace Task1_2_
{
    class Person
    {
        public string Name { get; set; }
        public int Age { get; set; }
        public IEnumerable<string> PhoneNumbers { get; set; }
        public Person(string name, int age, IEnumerable<string> phoneNumbers)
        {
            Name = name;
            Age = age;
            PhoneNumbers = phoneNumbers;
        }
    }
    class Program
    {
        static void Main(string[] args)
        {
            List<Person> PersonList = new List<Person>
            {
                new Person("David", 16, new List<string>() { "0732155750",
"0941533395" }),
                new Person("Anton", 31, new List<string>() { "0732155750",
"0941533395" }),
                new Person("Sasha", 89, new List<string>() { "0952145758",
"0941533395" }),
                new Person("Ihor", 8, new List<string>() { "0732155750", "0941533395"
}),
                new Person("Ilia", 15, new List<string>() { "0732155750", "0941533395"
})
            };

            foreach (var Person in PersonList)
            {
                Console.WriteLine("\n Name: " + Person.Name + " " + "Age: " +
Person.Age);
            }

            Console.WriteLine("\n Using the AddRange Method");

            PersonList.AddRange(new List<Person>() {
```

```
        new Person("Bob", 49, new List<string>() { "0732155750", "0941533395"
    })),
    new Person("Jake", 18, new List<string>() { "0732155750", "0941533395"
    })));
```

```
    foreach (var Person in PersonList)
    {
        Console.WriteLine("\n Name: " + Person.Name + " " + "Age: " +
    Person.Age);
    }
```

```
    foreach (var item in PersonList)
    {
        Console.WriteLine("\n {0} phone numbers:", item.Name);
        foreach (var number in item.PhoneNumbers)
        {
            Console.WriteLine("\t -{0}", number);
        }
    }
}
```

Завдання 3, полягало в написанні програми котра випадково генерує рядок, котрий містить 100 символів, довжина стрічки 4, а всі символи великі.

Також програма має вилучити всі символи, що повторюються та починаються з літери «Z».

Для наповнення стрічки прописую рядки:

```
strings = new List<string>(100);
Random random = new Random();
string temp;

for (int i = 0; i < 100; i++)
{
    temp = "";
    for (int j = 0; j < 4; j++)
    {
        temp += (char)((random.Next() % 26) + 65);
    }

    strings.Add(temp);
}
```

Щоб на екрані відображувалась кількість елементів до фільтрації та після:

```
Console.WriteLine("Amount of symbols before filtering: {0}",
strings.Count);
```

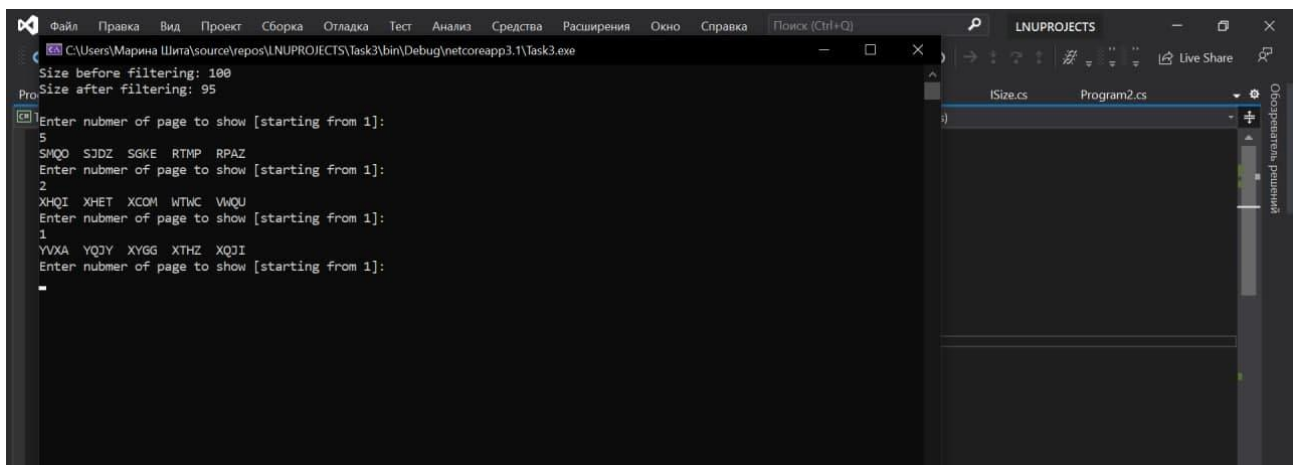
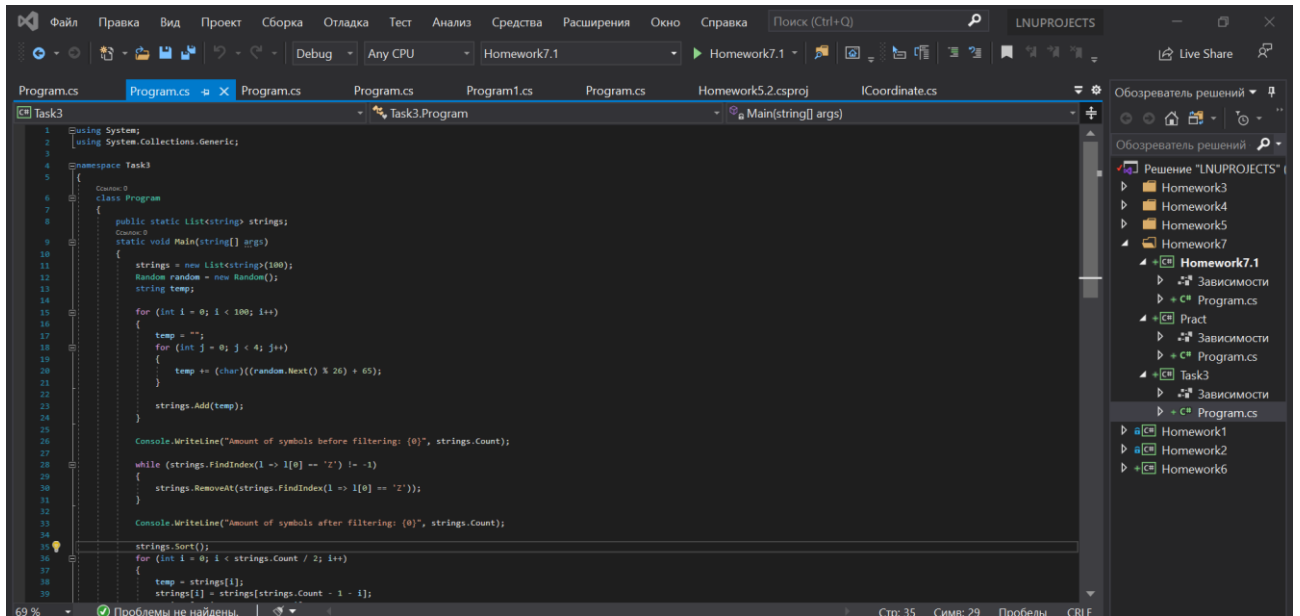
```
while (strings.FindIndex(l => l[0] == 'Z') != -1)
{
    strings.RemoveAt(strings.FindIndex(l => l[0] == 'Z'));
}
```

```
Console.WriteLine("Amount of symbols after filtering: {0}", strings.Count);
```

За сортування відповідають рядки:

```
strings.Sort();
for (int i = 0; i < strings.Count / 2; i++)
{
    temp = strings[i];
    strings[i] = strings[strings.Count - 1 - i];
    strings[strings.Count - 1 - i] = temp;
}
```

Також необхідно було створити метод DisplayPage(int pageNumber) при виклику якого на консоль буде виводитись відповідна сторінка оновленого списку (кількість елементів на сторінці константна: 5).



Код до завдання 3:

```
using System;
using System.Collections.Generic;

namespace Task3
{
    class Program
    {
        public static List<string> strings;
        static void Main(string[] args)
        {
            strings = new List<string>(100);
            Random random = new Random();
            string temp;

            for (int i = 0; i < 100; i++)
            {
                temp = "";
                for (int j = 0; j < 4; j++)
                {
                    temp += (char)((random.Next() % 26) + 65);
                }

                strings.Add(temp);
            }

            Console.WriteLine("Amount of symbols before filtering: {0}",
strings.Count);

            while (strings.FindIndex(l => l[0] == 'Z') != -1)
            {
                strings.RemoveAt(strings.FindIndex(l => l[0] == 'Z'));
            }

            Console.WriteLine("Amount of symbols after filtering: {0}", strings.Count);

            strings.Sort();
            for (int i = 0; i < strings.Count / 2; i++)
            {
                temp = strings[i];
                strings[i] = strings[strings.Count - 1 - i];
                strings[strings.Count - 1 - i] = temp;
            }

            while (true)
            {
```



```

        Console.WriteLine("\nEnter nubmer of page: ");
        if (!int.TryParse(Console.ReadLine(), out int pageNumber))
            break;
        pageNumber--;
        DisplayPage(pageNumber);
    }

    return;
}

public static void DisplayPage(int pageNum)
{
    if (pageNum * 5 > strings.Count
        || pageNum < 0)
    {
        Console.Write("Error. Enter another page");
        return;
    }
    for (int i = pageNum * 5; !(i >= (pageNum * 5) + 5
        || i >= strings.Count); i++)
    {
        Console.Write(strings[i] + " ");
    }

    return;
}
}
}

```

Висновок: на цій лабораторній роботі, я ознайомилась з поняттям колекції та навчилася їх застосовувати.