**МІНІСТЕРСТВО**

**ОСВІТИ ТА НАУКИ УКРАЇНИ**

**Національний технічний університет України**

**«Київський політехнічний інститут імені Ігоря Сікорського»**

**Кафедра інформаційних систем та технологій**

**Звіт**

**з лабораторної роботи No 3**

**«**Колекції C#. Списки. Словники. JSON. LINQ**»**

**з дисципліни**

**«Алгоритми і структури данних»**

**Варіант No29**

**Перевірив:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**  **Виконав:**

**Студент гр. ІС-12 ФІОТ**

**Хіврич Володимир**

**Київ 2022**

**Лабораторна робота №3**

**Теорія**

Колекції c#

<http://bit.ly/2TuUhEb>

Списки

<http://bit.ly/2TcniWd>

Словники

<https://metanit.com/sharp/tutorial/4.9.php>

<http://bit.ly/2IbQmXO>

JSON

<https://www.newtonsoft.com/json>

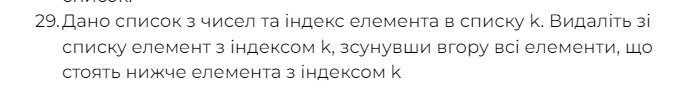
LINQ. Ця тема більш наближена до advanced рівня програмування. Тема дуже об’ємна, тому раджу пошукати про це інфу самостійно. На цьому сайті є цілий розділ присвячений темі LINQ

<https://metanit.com/sharp/tutorial/15.1.php>

<https://coderoad.ru/510531/%D0%92%D0%B0%D1%88%D0%B8-%D0%9B%D1%8E%D0%B1%D0%B8%D0%BC%D1%8B%D0%B5-LINQ-to-Objects-%D0%97%D0%B0%D0%BF%D1%80%D0%BE%D1%81%D0%BE%D0%B2>

**Завдання 1.**

Написати програму згідно отриманого завдання використовуючи колекції C#.

****

**Лістинг програми**

using System;

using System.Collections.Generic;

namespace lab3

{

class Program

{

static void Main(string[] args)

{

List<int> numbers = new List<int>();

for (int i = 1; i < 15; i++)

{

numbers.Add(i);

}

Console.WriteLine("Before remove");

printList(numbers);

Console.WriteLine();

int k = 7;

Console.WriteLine("After remove");

printList(removeNumber(k, numbers));

}

// Удаляет элемент по индексу, а все элементы, которые были ниже удаленного, перемещает в начало списка.

// Возвращает измененную коллекцию

private static List<int> removeNumber(int index, List<int> numbers)

{

int nextIndex = index + 1;

List<int> belowNum = new List<int>();

for (int i = nextIndex; i < numbers.Count; i++)

belowNum.Add(numbers[i]);

numbers.RemoveAt(index);

for (int i = 0; i < index; i++)

belowNum.Add(numbers[i]);

return belowNum;

}

private static void printList(List<int> numbers)

{

foreach (var item in numbers)

{

Console.Write(item + " ");

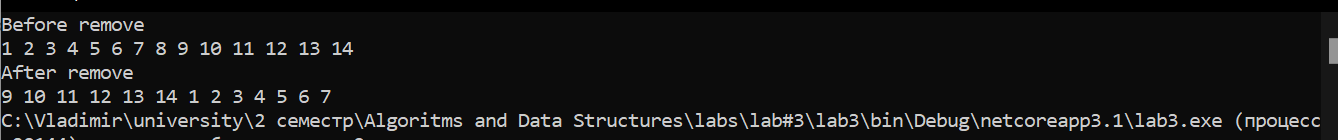
}

}

}

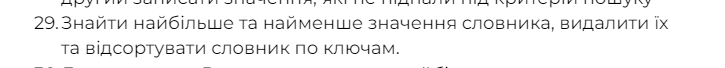
}

**Результат виконання (якщо індекс = 7)**

****

**Завдання 2**

Написати програму згідно виданого завдання використовуючи словники Dictionary в C#. Якщо результатом виконання програми є словник, **зберегти** цей результат у JSON файл

****

**Лістинг програми**

using System;

using System.Collections.Generic;

using System.IO;

using System.Linq;

using Newtonsoft.Json;

using Newtonsoft.Json.Linq;

namespace lab3.\_2

{

class Program

{

static void Main(string[] args)

{

Dictionary<string, int> numbers = new Dictionary<string, int>()

{

{ "c", 56},

{ "g", 24},

{ "a", 47},

{ "e", 1},

{ "y", 44},

{ "z", 9},

{ "b", 27},

{ "d", 55},

{ "x", 97},

{ "f", -12}

};

Console.WriteLine("Our dictionaty before operatoins");

printDictionary(numbers);

deleteMinAndMaxNumber(numbers);

numbers = sortDictionaryByKey(numbers);

Console.WriteLine("Our dictionaty after operatoins");

printDictionary(numbers);

string filePath = "sortedDictionary";

jsonSerialize(numbers, filePath);

}

private static void deleteMinAndMaxNumber(Dictionary<string, int> numbers)

{

int min = int.MaxValue;

int max = int.MinValue;

string keyOfMinNum = "", keyOfMaxNum = "";

foreach (var item in numbers)

{

if (item.Value < min)

{

min = item.Value;

keyOfMinNum = item.Key;

}

else if (item.Value > max)

{

max = item.Value;

keyOfMaxNum = item.Key;

}

}

Console.WriteLine("The biggest number is " + max);

Console.WriteLine("The least number is " + min);

numbers.Remove(keyOfMinNum);

numbers.Remove(keyOfMaxNum);

}

private static void printDictionary(Dictionary<string, int> numbers)

{

foreach (var item in numbers)

{

Console.WriteLine(item.Key + " -> " + item.Value);

}

}

private static Dictionary<string, int> sortDictionaryByKey(Dictionary<string, int> numbers)

{

return numbers.OrderBy(x => x.Key).ToDictionary(x => x.Key, x => x.Value);

}

private static void jsonSerialize(object data, string filePath)

{

JsonSerializer jsonSerializer = new JsonSerializer();

if (File.Exists(filePath)) File.Delete(filePath);

StreamWriter sw = new StreamWriter(filePath);

JsonWriter jsonWriter = new JsonTextWriter(sw);

jsonSerializer.Serialize(jsonWriter, data);

jsonWriter.Close();

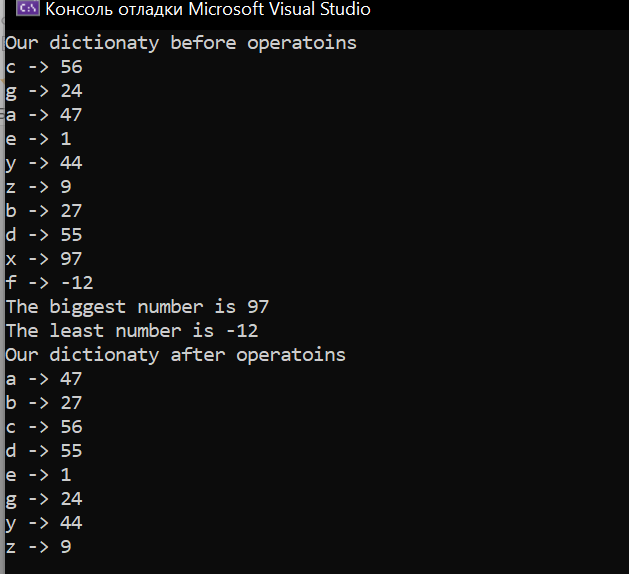
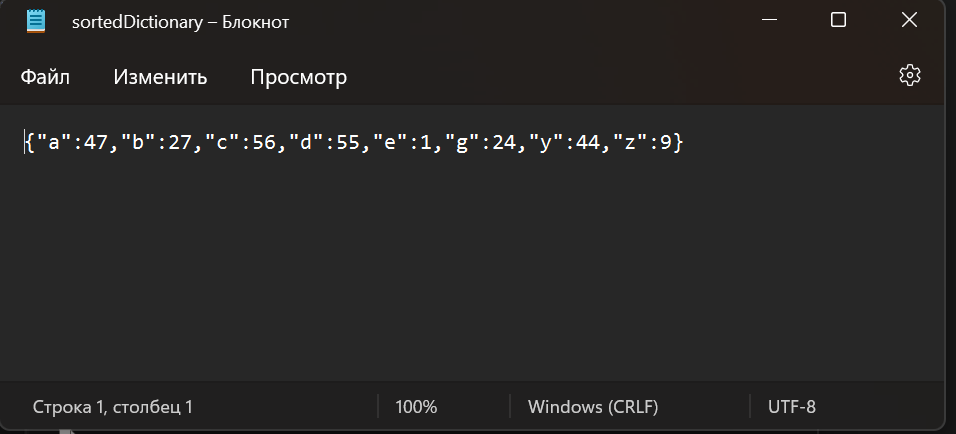
sw.Close();

}

}

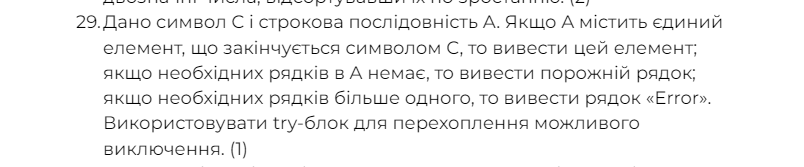
}

**Результат виконання програми**

****

**Завдання 3.**

Написати програму згідно виданого завдання використовуючи лише LINQ методи. В кінці завдання в дужках наведена підказка, які методи LINQ могли б вам допомогти у вирішення задачі

****

**Лістинг програми**

using System;

using System.Collections.Generic;

using System.Linq;

namespace lab3.\_3{

class Program

{

static void Main(string[] args)

{

char C = 's';

string str1 = "mercedes apple BMW Alibaba Amazon"; // consisits one

string str2 = "mercedes apple BMW Alibaba Airpods"; // consisits two

string str3 = "apple BMW Alibaba"; /// consisits none

string[] arr = str3.Split(" ");

List<string> list = new List<string>(arr);

int counter = 0;

string element = "";

foreach (var item in list)

{

if (item.Last() == C)

{

counter++;

element = item;

}

}

if (counter == 1)

{

Console.WriteLine(element);

}

else if (counter == 0)

{

Console.WriteLine("\*Empty line, as there is no words ending with \"" + C + "\"\*");

}

else

{

Console.WriteLine("Error");

}

}

}

}

**Результат виконання програми**

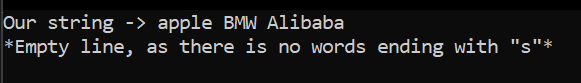
1. **Якщо слово одне в рядку**

****

1. **Якщо слів більше ніж одне**

****

1. **Якщо слів взагалі немає**

****