**Міністерство Освіти І НАУКИ України**

**Національний університет "Львівська політехніка"**

**КНІТ**

Кафедра **ПЗ**

### ЗВІТ

До лабораторної роботи № 3

**З дисципліни:** *“Алгоритми та структури даних”*

**На тему:** *“Метод сортування Шелла”*

**Лектор:**

доц. каф. ПЗ

Коротєєва Т.О.

**Виконала:**

ст. групи ПЗ-26

Матолінець Л. А.

**Прийняв:**

асистент каф. ПЗ

Симець І.І.

« \_\_\_\_ » \_\_\_\_\_\_\_\_ 2022 р.

∑= \_\_\_\_\_ .

Львів – 2022

**Тема роботи:** Метод сортування Шелла**.**

**Мета роботи**: Вивчити [алгоритм](http://vns.lp.edu.ua/mod/glossary/showentry.php?courseid=472&eid=19102&displayformat=dictionary" \o "Глосарій: Алгоритм) сортування Шелла. Здійснити програмну реалізацію [алгоритм](http://vns.lp.edu.ua/mod/glossary/showentry.php?courseid=472&eid=19102&displayformat=dictionary" \o "Глосарій: Алгоритм)у сортування Шелла. Дослідити швидкодію [алгоритм](http://vns.lp.edu.ua/mod/glossary/showentry.php?courseid=472&eid=19102&displayformat=dictionary" \o "Глосарій: Алгоритм)у сортування Шелла.

**TЕОРЕТИЧНІ ВІДОМОСТІ**

***Покроковий опис роботи алгоритму сортування Шелла.***

**Алгоритм SH** Задано одновимірний масив елементів *R1 ,R2 ,…,Rn.* Даний алгоритм реорганізує масив у висхідному порядку, порівнюючи елементи масиву, між якими є відстань gap, поки ця змінна не набуде значення 1 включно.

S1. Цикл за індексом проходження i; i = gap. Повторювати кроки S2-S4 при i=1…n.

S2. Зафіксувати значення змінної j = i;

S3. . Цикл перевірки умови: знаходимо елемент, на місце якого потрібно вставити поточний; якщо перший елемент з двох, відстань між якими дорівнює gap, є більший, то -> S4.

S4. Перестановка даних двох елементів.. Встановлення значення змінної j: j = j - gap.

S5. Кінець. Вихід.

**ІНДИВІДУАЛЬНЕ ЗАВДАННЯ**

**Варіант 3**

Задано одномірний масив дійсних чисел. До парних елементів масиву застосувати функцію: . Отриманий масив посортувати в порядку зростання.

**ХІД ВИКОНАННЯ**

Метод сортування Шелла:

using System;

using System.Diagnostics;

using System.Threading;

class HelloWorld {

static void Main() {

int Min = -50000;

int Max = 50000;

Console.WriteLine("Enter the size of your array: ");

int length = Convert.ToInt32(Console.ReadLine());

int[] unsortedArr = new int[length];

Random randNum = new Random();

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

{

unsortedArr[i] = randNum.Next(Min, Max);

}

int[] newArray = shellSort(unsortedArr);

}

static void swap(ref int a, ref int b)

{

var t = a;

a = b;

b = t;

}

static int[] shellSort(int[] array)

{

Stopwatch stopwatch = new Stopwatch();

stopwatch.Start();

        int n = arr.Length;

        for (int gap = n/2; gap >= 1; gap /= 2)

        {

            for (int i = gap; i < n; i += 1)

            {

                int temp = arr[i];

                int j = i;

                for (j >= gap && arr[j - gap] > temp; j -= gap)

                    arr[j] = arr[j - gap];

                arr[j] = temp;

            }

        }

        return 0;

    }

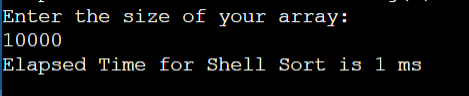
stopwatch.Stop();

Console.WriteLine("Elapsed Time for Shell Sort is {0} ms", stopwatch.ElapsedMilliseconds);

return array;

}

}



*Рис.1 Результат виконання програми*

Вихідний код виконання індивідуального завдання:

//myArray.cs

using System;

using System.Collections.Generic;

using System.Linq;

namespace SortingApp

{

public class MyArray

{

protected double[] array = new double[0];

public MyArray() { }

public double[] getList() { return array; }

public void addElement(double item) {

Array.Resize(ref array, array.Length + 1);

array[array.Length-1] = item;

}

public void transformArray()

{

for(int i = 0; i < this.array.Length; i++) {

if ((array[i]\*10) % 2 == 0) {

double valToChange = array[i];

array[i] = Math.Sqrt(Math.Abs(valToChange - 10));

};

}

}

}

}

// MainWindow.xaml.cs

using System;

using System.Collections.Generic;

using System.Windows;

using System.Windows.Controls;

using System.Windows.Documents;

namespace SortingApp

{

/// <summary>

/// Interaction logic for MainWindow.xaml

/// </summary>

public partial class MainWindow : Window

{

MyArray digitsArr = new MyArray();

public MainWindow()

{

InitializeComponent();

}

private void addArray\_Click(object sender, RoutedEventArgs e)

{

String arrField = arrayField.Text;

String[] strings = arrField.Split(" ; ");

foreach (var item in strings)

{

double x = Convert.ToDouble(item);

digitsArr.addElement(x);

}

}

static void swap(ref double a, ref double b)

{

var t = a;

a = b;

b = t;

}

static double[] shellSort(double[] array, TextBox textBox)

{

textBox.Text = "";

String line = "";

int gap = array.Length / 2;

int counter = 0;

while (gap >= 1)

{

for (int i = gap; i < array.Length; i++)

{

line += counter + ": ";

int j = i;

foreach (var item in array)

{

line += Math.Round(item, 2);

line += " ; ";

}

while ((j >= gap) && (array[j - gap] > array[j]))

{

swap(ref array[j], ref array[j - gap]);

j = j - gap;

}

line += "\n\n";

textBox.Text += line;

line = "";

counter++;

}

gap /= 2;

}

return array;

}

private void Button\_Click(object sender, RoutedEventArgs e)

{

digitsArr.transformArray();

double[] arr = digitsArr.getList();

double[] sortedArray = shellSort(arr, stepsArea);

String result = "";

foreach (var item in sortedArray)

{

result += Convert.ToString(Math.Round(item, 1));

result += " ; ";

}

resultField.Content = result;

}

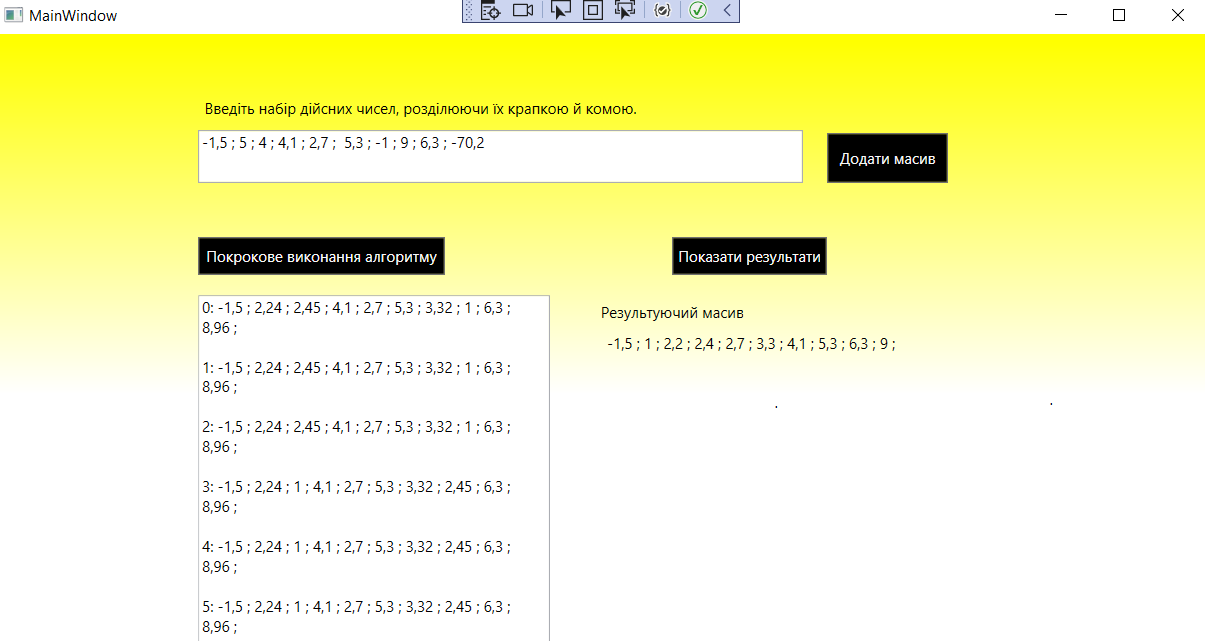
private void showSteps\_Click(object sender, RoutedEventArgs e)

{

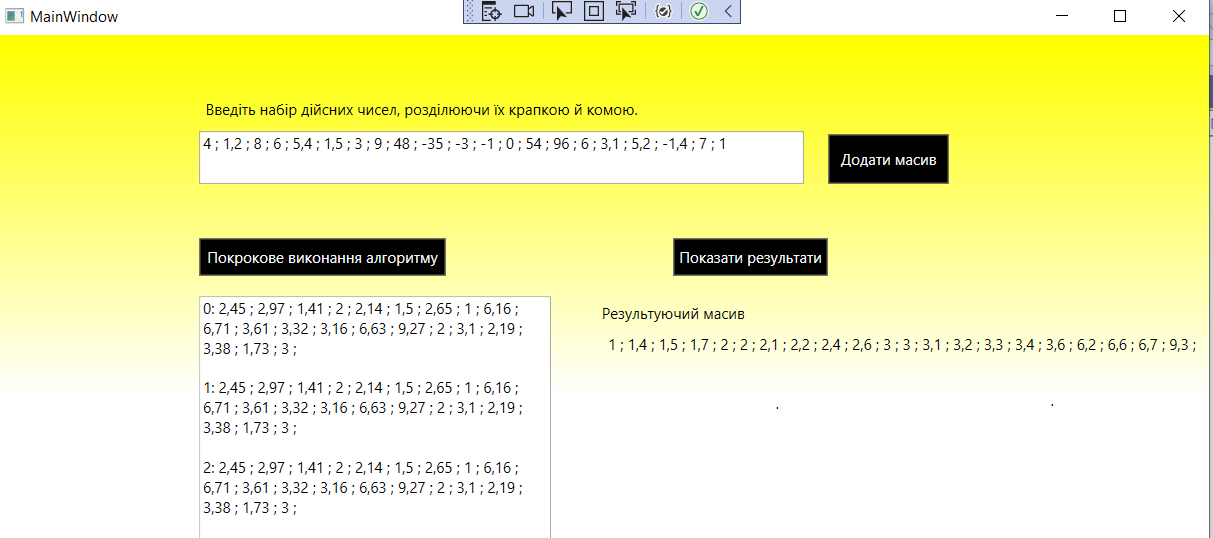
}

}

}



*Рис. 2 Результат сортування 10ти елементів*



*Рис. 2 Результат сортування 20ти елементів*

**Висновок**

На цій лабораторній роботі я познайомилася з алгоритмом сортування Шелла, який є покращеним варіантом сортування вставкою. Складність такого алгоритму – О(n1.5). Також в межах цієї теми я виконала індивідуальне завдання – сортування одновимірного масиву дійсних чисел, застосовуючи спеціальну умову для всіх парних.