Національний технічний університет України «КПІ ім. І. Сікорського»

Факультет інформатики і обчислювальної техніки

Кафедра автоматизованих систем обробки інформації та управління

**Лабораторна робота №5**

**з дисципліни**

**«ОРПЗ .NET»**

на тему

# «Породжуючі шаблони»

Виконав

студент групи ІС-03

Ніколенко Гліб

Київ – 2022

Варіант 11:



Я використав паттерн «Стратегія», оскільки він чудово вирішує моє питання з вибором способу сортування.

Код програми:

using System;

using System.Collections.Generic;

namespace lab5

{

public interface IStrategy

{

int[] DoAlhorithm(int[] data);

}

class ConcreteAlhorithmBubble : IStrategy

{

public int[] DoAlhorithm(int[] data)

{

var len = data.Length;

for (var i = 1; i < data.Length; i++)

{

for (var j = 0; j < data.Length - i; j++)

{

if (data[j] > data[j + 1])

{

var temp = data[j];

data[j] = data[j + 1];

data[j + 1] = temp;

}

}

}

return data;

}

}

class ConcreteAlhorithmInsertionSort : IStrategy

{

public int[] DoAlhorithm(int[] data)

{

for (var i = 1; i < data.Length; i++)

{

var key = data[i];

var j = i;

while ((j > 1) && (data[j - 1] > key))

{

var temp = data[j - 1];

data[j - 1] = data[j];

data[j] = temp;

j--;

}

data[j] = key;

}

return data;

}

}

class Context

{

private IStrategy \_strategy;

public Context()

{ }

public Context(IStrategy strategy)

{

this.\_strategy = strategy;

}

public void SetStrategy(IStrategy strategy)

{

this.\_strategy = strategy;

}

public void DoSomeBusinessLogic(int[] data)

{

int[] result = this.\_strategy.DoAlhorithm(data);

for (int i = 0; i < data.Length; i++)

{

Console.WriteLine(result[i]);

}

}

}

class Program

{

static void Main(string[] args)

{

const int n = 10;

int[] arr = new int[n] { 1, 12, 65, 1245, 567435, 123, 9, 213, 31, 753 };

var context = new Context();

Console.WriteLine("Please, choose the alhorithm you want to sort the array: \n1)If you want Bubble Sort press 1 \n2)If you want Insertion Sort press 2\n");

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

if (variant == 1)

{

Console.WriteLine("Client: Strategy is set to Bubble sorting.");

context.SetStrategy(new ConcreteAlhorithmBubble());

context.DoSomeBusinessLogic(arr);

}

else if (variant == 2)

{

Console.WriteLine("Client: Strategy is set to Insert sorting.");

context.SetStrategy(new ConcreteAlhorithmInsertionSort());

context.DoSomeBusinessLogic(arr);

}

else

{

Console.WriteLine("There is no such variant...");

}

Console.WriteLine("Great, next choose what element you want: \n1)Max element - press 1 \n2)Min element - press 2\n");

int variant2 = Convert.ToInt32(Console.ReadLine());

if (variant2 == 1)

{

Console.WriteLine("Client wants max element.");

int maxEl = arr[0];

for (int i = 0; i < arr.Length; i++)

{

if (arr[i] > maxEl)

{

maxEl = arr[i];

}

}

Console.WriteLine(maxEl);

}

else if (variant2 == 2)

{

Console.WriteLine("Client wants min element.");

int minEl = arr[0];

for (int i = 0; i < arr.Length; i++)

{

if (arr[i] < minEl)

{

minEl = arr[i];

}

}

Console.WriteLine(minEl);

}

else

{

Console.WriteLine("There is no such variant...");

}

}

}

}

Програма працює наступник чином:

1)Користувач обирає з 2 варіантів сортування той, який йому подобається більше.

2)За допомогою виклику SetStrategy – я викликаю той Алгоритм, який потрібен користувачеві.

3)Користувач обирає, який елемент він хоче побачити, мінімальний чи максимальний.

4)Виводжу бажаний результат

Результат роботи на прикладі вибору сортування вставкою та максимального елемента:

