M, nМинистерство образования и науки Украины

Национальный технический университет Украины

«Киевский политехнический институт им. Игоря Сикорского»

Факультет информатики и вычислительной техники

Кафедра вычислительной техники

Лабораторная работа №6

по дисциплине «Инженерия программного обеспечения»

на тему «Шаблоны поведения.

Шаблоны Strategy, Chain Of Responsibility, Visitor»

ВЫПОЛНИЛА:

студентка 2-го курса

ФИВТ гр. ИВ-71

Молчанова В.С.

Зачетка – 7110

Вариант – 0

ПРОВЕРИЛ:

Старший преподаватель

к.т.н., с. н. с.

Антонюк А.И.

Киев - 2018

**1. Цель работы:**

Изучение шаблонов поведения. Получение базовых навыков применения шаблонов Strategy, Chain of Responsibility и Visitor.

**2. Вариант задания 7110 % 10 = 0:**

0. Определить спецификации класса, который содержит массив целых чисел и метод его сортировки. Обеспечить возможность динамического изменения алгоритма и направления сортировки путем внешней параметризации.

**3. Код программы**

package com.lab111.labwork6;

/\*\*

\* Класс реализующий конкретную стратегию, должен наследовать этот интерфейс

\* Класс контекста использует этот интерфейс для вызова конкретной стратегии

\*

\* @author Varvara Molchanova

\*/

public interface Strategy {

/\*\*

\* Сотрирует массив по определенному алгоритму

\*

\* @param mas - сортируемый массив

\*/

public void sort(Object[] mas);

}

package com.lab111.labwork6;

/\*\*

\* Реализуем алгоритм с использованием интерфейса стратегии

\*

\* @author Varvara Molchanova

\*/

public class SortOne implements Strategy {

@Override

public void sort(Object[] mas) {

System.out.println("Отсортировано первым");

}

}

package com.lab111.labwork6;

/\*\*

\* Реализуем алгоритм с использованием интерфейса стратегии

\*

\* @author Varvara Molchanova

\*/

public class SortTwo implements Strategy{

@Override

public void sort(Object[] mas) {

System.out.println("Отсортировано вторым");

}

}

package com.lab111.labwork6;

/\*\*

\* Класс контекста использующий интерфейс стратегии

\*

\* @author Varvara Molchanova

\*/

public class Context {

// Хранимый алгоритм

private Strategy strategy;

// Constructor

public Context() {

}

/\*\*

\* Set new strategy

\*/

public void setStrategy(Strategy strategy) {

this.strategy = strategy;

}

/\*\*

\* Do the strategy algorithm

\*/

public void sortByStrategy() {

strategy.sort(new Object[5]);

}

}

package com.lab111.labwork6;

/\*\*

\* Only workclass.

\* Realise pattern "Strategy". Its a client part.

\*

\* @author Varvara Molchanova

\*/

public class WorkClass {

public static void main(String[] args) {

Context context = new Context();

context.setStrategy(new SortOne());

context.sortByStrategy();

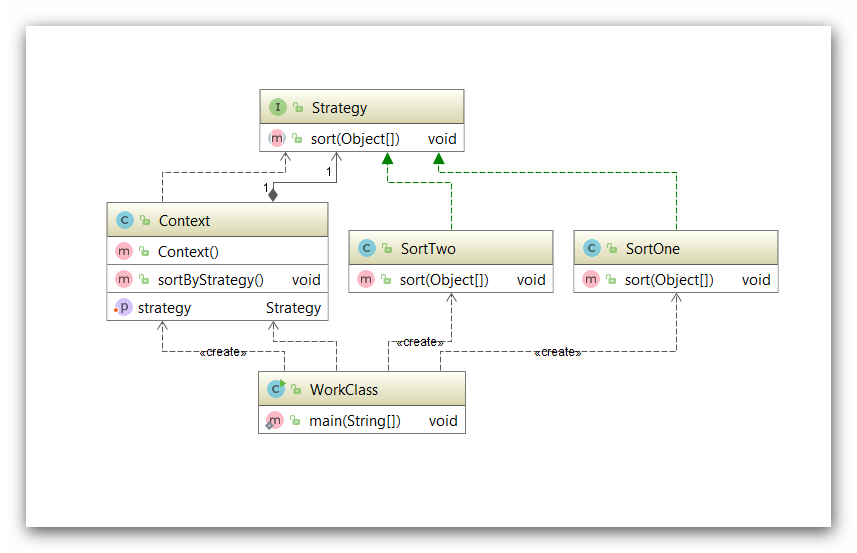
context.setStrategy(new SortTwo());

context.sortByStrategy();

}

}

**4. Диаграмма классов**

****

**5. Результаты тестирования программы**

Отсортировано первым

Отсортировано вторым

Process finished with exit code 0

**6. Выводы**

Ознакомились с темой лабораторной работы.

Приобретены соответствующие навыки работы.

Разработаны соответствующую тестовую программу.

Результаты успешной работы тестовой программы, приведенные выше, подтверждают правильность выбранных решений. Конечная цель работы достигнута.