Національний технічний університет України

«Київський політехнічний інститут»

Факультет інформатики та обчислювальної техніки

Кафедра обчислювальної техніки

Лабораторна робота № 6

з дисципліни: «Інженерія програмного забезпечення»

Тема: «Шаблони поведінки. Шаблони Strategy, Chain of Responsibility та Visitor»

Виконав:

студент групи ІО-33

Заколенко Роман

Перевірив:

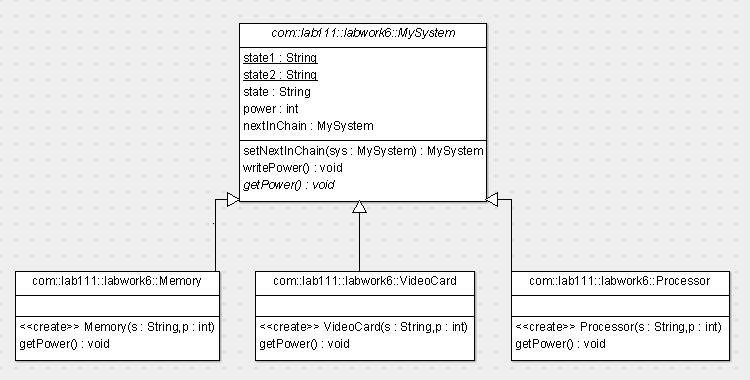
Антонюк А.І.

2014

**Мета роботи:** Вивчення шаблонів поведінки. Отримання базових навичок з застосування шаблонів Strategy, Chain of Responsibility та Visitor.

**Варіант завдання - 8 (3308 mod 10 = 8) :**

Визначити специфікації класів, що реалізують елементи структури комп’ютера (процесор, пам'ять відео карта тощо). Реалізувати механізм додаткових операцій над структурою комп’ютера без зміни її елементів. В якості ілюстрації такого механізму розробити операцію визначення потужності, що потребляє комп’ютер.

**UML-діаграма**

**Лістинг**

**package** com.lab111.labwork6;

/\*\*

\* Abstract class MySystem

\*

\* **@author** Roman Zakolenko

\* **@version** 6.8

\*/

**public** **abstract** **class** MySystem {

/\*\*

\* The static field state1

\*/

**public** **static** String *state1* = "OFF";

/\*\*

\* The static field state2

\*/

**public** **static** String *state2* = "ON";

/\*\*

\* The field state

\*/

**public** String state;

/\*\*

\* The field power which contained power value of object

\*/

**public** **int** power;

/\*\*

\* The field nextInChain contained link to the next object in chain of

\* responsibility

\*/

**protected** MySystem nextInChain;

/\*\*

\* **@param** sys

\* assigned with the field nextInChain

\* **@return** link of object MySystem

\*/

**public** MySystem setNextInChain(MySystem sys) {

nextInChain = sys;

**return** sys;

}

/\*\*

\* The method writePower()

\*/

**public** **void** writePower() {

**if** (state.equals("ON")) {

getPower();

}

**if** (nextInChain != **null**) {

nextInChain.writePower();

}

}

/\*\*

\* The abstract method getPower;

\*/

**abstract** **protected** **void** getPower();

}

**package** com.lab111.labwork6;

/\*\*

\* Class Processor extends abstract class MySystem

\* **@author** Roman Zakolenko

\* **@version** 6.8

\*/

**public** **class** Processor **extends** MySystem {

/\*\*

\* **@param** s assigned with the field state

\* **@param** p assigned with the field power

\*/

**public** Processor(String s, **int** p) {

**this**.state = s;

**this**.power = p;

}

/\* (non-Javadoc)

\* @see com.lab111.labwork6.MySystem#getPower()

\*/

@Override

**protected** **void** getPower() {

System.*out*.println("Power of processor : " + **this**.power);

}

}

**package** com.lab111.labwork6;

/\*\*

\* Class VideoCard extends abstract class MySystem

\* **@author** Roman Zakolenko

\* **@version** 6.8

\*/

**public** **class** VideoCard **extends** MySystem {

/\*\*

\* **@param** s assigned with the field state

\* **@param** p assigned with the field power

\*/

**public** VideoCard(String s, **int** p) {

**this**.state = s;

**this**.power = p;

}

/\* (non-Javadoc)

\* @see com.lab111.labwork6.MySystem#getPower()

\*/

@Override

**protected** **void** getPower() {

System.*out*.println("Power of video card : " + **this**.power);

}

}

**package** com.lab111.labwork6;

/\*\*

\* Class Memory extends abstract class MySystem

\*

\* **@author** Roman Zakolenko

\* **@version** 6.8

\*/

**public** **class** Memory **extends** MySystem {

/\*\*

\* **@param** s

\* assigned with the field state

\* **@param** p

\* assigned with the field power

\*/

**public** Memory(String s, **int** p) {

**this**.state = s;

**this**.power = p;

}

/\*

\* (non-Javadoc)

\*

\* @see com.lab111.labwork6.MySystem#getPower()

\*/

@Override

**protected** **void** getPower() {

System.*out*.println("Power of memory : " + **this**.power);

}

}

**package** com.lab111.labwork6;

/\*\*

\* The class Lab6 contained the main method

\*

\* **@author** Roman Zakolenko

\* **@version** 6.8

\*/

**public** **class** Lab6 {

**public** **static** **void** main(String[] args) {

MySystem processor = **new** Processor(MySystem.*state2*, 100);

MySystem videoCard = processor.setNextInChain(**new** VideoCard(

MySystem.*state1*, 80));

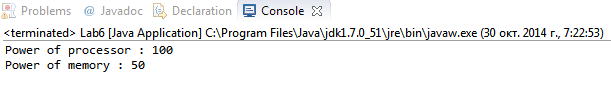
MySystem memory = videoCard.setNextInChain(**new** Memory(MySystem.*state2*,

50));

processor.writePower();

}

}

**Результати роботи програми**

**Документація**

## C

[com.lab111.labwork6](file:///D:\ФІОТ\IO-33\2%20курс\IPZ\IPZ_lab1_var5\doc\com\lab111\labwork6\package-summary.html) - package com.lab111.labwork6

## G

[**getPower()**](file:///D:\ФІОТ\IO-33\2%20курс\IPZ\IPZ_lab1_var5\doc\com\lab111\labwork6\Memory.html#getPower()) - Method in class com.lab111.labwork6.[Memory](file:///D:\ФІОТ\IO-33\2%20курс\IPZ\IPZ_lab1_var5\doc\com\lab111\labwork6\Memory.html)

[**getPower()**](file:///D:\ФІОТ\IO-33\2%20курс\IPZ\IPZ_lab1_var5\doc\com\lab111\labwork6\MySystem.html#getPower()) - Method in class com.lab111.labwork6.[MySystem](file:///D:\ФІОТ\IO-33\2%20курс\IPZ\IPZ_lab1_var5\doc\com\lab111\labwork6\MySystem.html)

The abstract method getPower;

[**getPower()**](file:///D:\ФІОТ\IO-33\2%20курс\IPZ\IPZ_lab1_var5\doc\com\lab111\labwork6\Processor.html#getPower()) - Method in class com.lab111.labwork6.[Processor](file:///D:\ФІОТ\IO-33\2%20курс\IPZ\IPZ_lab1_var5\doc\com\lab111\labwork6\Processor.html)

[**getPower()**](file:///D:\ФІОТ\IO-33\2%20курс\IPZ\IPZ_lab1_var5\doc\com\lab111\labwork6\VideoCard.html#getPower()) - Method in class com.lab111.labwork6.[VideoCard](file:///D:\ФІОТ\IO-33\2%20курс\IPZ\IPZ_lab1_var5\doc\com\lab111\labwork6\VideoCard.html)

## 

## L

[Lab6](file:///D:\ФІОТ\IO-33\2%20курс\IPZ\IPZ_lab1_var5\doc\com\lab111\labwork6\Lab6.html) - Class in [com.lab111.labwork6](file:///D:\ФІОТ\IO-33\2%20курс\IPZ\IPZ_lab1_var5\doc\com\lab111\labwork6\package-summary.html)

The class Lab6 contained the main method

[**Lab6()**](file:///D:\ФІОТ\IO-33\2%20курс\IPZ\IPZ_lab1_var5\doc\com\lab111\labwork6\Lab6.html#Lab6()) - Constructor for class com.lab111.labwork6.[Lab6](file:///D:\ФІОТ\IO-33\2%20курс\IPZ\IPZ_lab1_var5\doc\com\lab111\labwork6\Lab6.html)

## M

[**main(String[])**](file:///D:\ФІОТ\IO-33\2%20курс\IPZ\IPZ_lab1_var5\doc\com\lab111\labwork6\Lab6.html#main(java.lang.String[])) - Static method in class com.lab111.labwork6.[Lab6](file:///D:\ФІОТ\IO-33\2%20курс\IPZ\IPZ_lab1_var5\doc\com\lab111\labwork6\Lab6.html)

[Memory](file:///D:\ФІОТ\IO-33\2%20курс\IPZ\IPZ_lab1_var5\doc\com\lab111\labwork6\Memory.html) - Class in [com.lab111.labwork6](file:///D:\ФІОТ\IO-33\2%20курс\IPZ\IPZ_lab1_var5\doc\com\lab111\labwork6\package-summary.html)

Class Memory extends abstract class MySystem

[**Memory(String, int)**](file:///D:\ФІОТ\IO-33\2%20курс\IPZ\IPZ_lab1_var5\doc\com\lab111\labwork6\Memory.html#Memory(java.lang.String,%20int)) - Constructor for class com.lab111.labwork6.[Memory](file:///D:\ФІОТ\IO-33\2%20курс\IPZ\IPZ_lab1_var5\doc\com\lab111\labwork6\Memory.html)

[MySystem](file:///D:\ФІОТ\IO-33\2%20курс\IPZ\IPZ_lab1_var5\doc\com\lab111\labwork6\MySystem.html) - Class in [com.lab111.labwork6](file:///D:\ФІОТ\IO-33\2%20курс\IPZ\IPZ_lab1_var5\doc\com\lab111\labwork6\package-summary.html)

Abstract class MySystem

[**MySystem()**](file:///D:\ФІОТ\IO-33\2%20курс\IPZ\IPZ_lab1_var5\doc\com\lab111\labwork6\MySystem.html#MySystem()) - Constructor for class com.lab111.labwork6.[MySystem](file:///D:\ФІОТ\IO-33\2%20курс\IPZ\IPZ_lab1_var5\doc\com\lab111\labwork6\MySystem.html)

## N

[**nextInChain**](file:///D:\ФІОТ\IO-33\2%20курс\IPZ\IPZ_lab1_var5\doc\com\lab111\labwork6\MySystem.html#nextInChain) - Variable in class com.lab111.labwork6.[MySystem](file:///D:\ФІОТ\IO-33\2%20курс\IPZ\IPZ_lab1_var5\doc\com\lab111\labwork6\MySystem.html)

The field nextInChain contained link to the next object in chain of responsibility

## P

[**power**](file:///D:\ФІОТ\IO-33\2%20курс\IPZ\IPZ_lab1_var5\doc\com\lab111\labwork6\MySystem.html#power) - Variable in class com.lab111.labwork6.[MySystem](file:///D:\ФІОТ\IO-33\2%20курс\IPZ\IPZ_lab1_var5\doc\com\lab111\labwork6\MySystem.html)

The field power which contained power value of object

[Processor](file:///D:\ФІОТ\IO-33\2%20курс\IPZ\IPZ_lab1_var5\doc\com\lab111\labwork6\Processor.html) - Class in [com.lab111.labwork6](file:///D:\ФІОТ\IO-33\2%20курс\IPZ\IPZ_lab1_var5\doc\com\lab111\labwork6\package-summary.html)

Class Processor extends abstract class MySystem

[**Processor(String, int)**](file:///D:\ФІОТ\IO-33\2%20курс\IPZ\IPZ_lab1_var5\doc\com\lab111\labwork6\Processor.html#Processor(java.lang.String,%20int)) - Constructor for class com.lab111.labwork6.[Processor](file:///D:\ФІОТ\IO-33\2%20курс\IPZ\IPZ_lab1_var5\doc\com\lab111\labwork6\Processor.html)

## S

[**setNextInChain(MySystem)**](file:///D:\ФІОТ\IO-33\2%20курс\IPZ\IPZ_lab1_var5\doc\com\lab111\labwork6\MySystem.html#setNextInChain(com.lab111.labwork6.MySystem)) - Method in class com.lab111.labwork6.[MySystem](file:///D:\ФІОТ\IO-33\2%20курс\IPZ\IPZ_lab1_var5\doc\com\lab111\labwork6\MySystem.html)

[**state**](file:///D:\ФІОТ\IO-33\2%20курс\IPZ\IPZ_lab1_var5\doc\com\lab111\labwork6\MySystem.html#state) - Variable in class com.lab111.labwork6.[MySystem](file:///D:\ФІОТ\IO-33\2%20курс\IPZ\IPZ_lab1_var5\doc\com\lab111\labwork6\MySystem.html)

The field state

[**state1**](file:///D:\ФІОТ\IO-33\2%20курс\IPZ\IPZ_lab1_var5\doc\com\lab111\labwork6\MySystem.html#state1) - Static variable in class com.lab111.labwork6.[MySystem](file:///D:\ФІОТ\IO-33\2%20курс\IPZ\IPZ_lab1_var5\doc\com\lab111\labwork6\MySystem.html)

The static field state1

[**state2**](file:///D:\ФІОТ\IO-33\2%20курс\IPZ\IPZ_lab1_var5\doc\com\lab111\labwork6\MySystem.html#state2) - Static variable in class com.lab111.labwork6.[MySystem](file:///D:\ФІОТ\IO-33\2%20курс\IPZ\IPZ_lab1_var5\doc\com\lab111\labwork6\MySystem.html)

The static field state2

## V

[VideoCard](file:///D:\ФІОТ\IO-33\2%20курс\IPZ\IPZ_lab1_var5\doc\com\lab111\labwork6\VideoCard.html) - Class in [com.lab111.labwork6](file:///D:\ФІОТ\IO-33\2%20курс\IPZ\IPZ_lab1_var5\doc\com\lab111\labwork6\package-summary.html)

Class VideoCard extends abstract class MySystem

[**VideoCard(String, int)**](file:///D:\ФІОТ\IO-33\2%20курс\IPZ\IPZ_lab1_var5\doc\com\lab111\labwork6\VideoCard.html#VideoCard(java.lang.String,%20int)) - Constructor for class com.lab111.labwork6.[VideoCard](file:///D:\ФІОТ\IO-33\2%20курс\IPZ\IPZ_lab1_var5\doc\com\lab111\labwork6\VideoCard.html)

## W

[**writePower()**](file:///D:\ФІОТ\IO-33\2%20курс\IPZ\IPZ_lab1_var5\doc\com\lab111\labwork6\MySystem.html#writePower()) - Method in class com.lab111.labwork6.[MySystem](file:///D:\ФІОТ\IO-33\2%20курс\IPZ\IPZ_lab1_var5\doc\com\lab111\labwork6\MySystem.html)

The method writePower()

**Висновки:**

В результаті виконання лабораторної роботи я вивчив три шаблони поведінки, а саме Strategy, Chain of Responsibility та Visitor, а також практично реалізував шаблон Chain of Responsibility.