Министерство образования и науки Российской Федерации

Новосибирский государственный технический университет

Кафедра прикладной математики

Лабораторная работа № 5

по дисциплине "Объектно-ориентированное программирование"

**Шаблоны**

Факультет: прикладной математики и информатики

Группа: ПМИ-41

Студент: Ромашев В.А.

Преподаватель: Тракимус Ю. В.

Новосибирск  
2016

1. Условие задачи

Преобразовать класс-контейнер, разработанный в лабораторной работе №3, в шаблон, так чтобы элементами контейнеров могли быть различные классы, разработанные в лабораторной работе №4.

1. Внутреннее представление данных

class Table <T> where T: Shape

{

public class list

{

public T data = default(T); // объект

public int key; // идентификатор

public list next = null; // следующий элемент таблицы

public list prev = null; // предыдущий элемент таблицы

}

public list Head = new list(); // указатель на начало таблицы

public void inTable(list r, T p, int k) // добавление в таблицу

public T Search(int k) // поиск элемента в таблице

public void printDirectOrder (Graphics G) // печать таблицы в прямом порядке

public void printReverseOrder(Graphics G) // печать таблицы в обратном порядке

}

1. Текст программы
   1. Table.CS

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Drawing;

using System.Windows.Forms;

namespace OOP\_LR4

{

class Table <T> where T: Shape

{

public class list

{

public T data = default(T); // объект

public int key; // идентификатор

public list next = null; // следующий элемент таблицы

public list prev = null; // предыдущий элемент таблицы

}

public list Head = new list(); // указатель на начало таблицы

public void inTable(list r, T p, int k) // добавление в таблицу

{

if (r.data == null)

{

r.data = p;

r.key = k;

Head.prev = r;

}

else

{

r.next = new list();

r.next.prev = r;

r.next.key = k;

r.next.data = p;

Head.prev = r.next;

}

}

public T Search(int k) // поиск в таблице

{

list r = Head;

while (r != null && r.key != k)

{

r = r.next;

}

if (r != null)

return r.data;

else

return default(T);

}

public void printDirectOrder (Graphics G) // печать таблицы в прямом порядке

{

list r = Head;

if (r.data == null)

throw new Exception("таблица пуста");

while (r != null)

{

r.data.draw(G);

r = r.next;

}

}

public void printReverseOrder(Graphics G) // печать таблицы в обратном порядке

{

list r = Head.prev;

if (r == null)

throw new Exception("таблица пуста");

do

{

r.data.draw(G);

r = r.prev;

}

while (r.next != null);

}

}

}