Министерство образования и науки РФ

Севастопольский государственный университет

Институт информационных технологий и управления в технических сетях

Лабораторная работа №4

**Перегрузка операторов.**

Выполнил:

ст.гр.ИСб-22д

Воронин И.Ю.

Проверил:

Тимофеев И.С.

Севастополь

2015

1.Цель работы

Приобретение практических навыков при написании объектно-ориентированных программ с использованием перегруженных операторов.

2.Вариант задания

Вариант 10

10) Куча, состоящая из вещественных чисел.

10.1) + как добавление нового элемента в список.

10.2) ! как получение минимального элемента

10.3) ++ как увеличение каждого элемента на 1.

3.Текст программы

Файл Program.cs:

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace OOP\_4

{

class Program

{

static void Main(string[] args)

{

Heap h = new Heap();

h = h + 9.2;

h = h + 6.2;

h = h + 4.2;

h++;

h++;

Console.WriteLine(!h);

Console.ReadKey();

}

}

}

Файл Heap.cs:

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace OOP\_4

{

class List{

public double count;

public List next;

public override string ToString()

{

return count.ToString();

}

}

class Heap

{

private List first;

private List last;

private int amount;

public Heap(){

first = null;

amount = 0;

}

public override string ToString(){

string double\_row = "";

List temp = first;

while(temp!=null){

temp = temp.next;

double\_row += temp.ToString();

double\_row += ' ';

}

return double\_row;

}

public static Heap operator+ (Heap heap, double \_count){

if (heap.first == null)

{

heap.first = new List();

heap.first.count = \_count;

heap.last = heap.first;

heap.last.next = null;

heap.amount++;

return heap;

}

else

{

heap.last.next = new List();

heap.last.next.count = \_count;

heap.last = heap.last.next;

heap.last.next = null;

heap.amount++;

return heap;

}

}

public static double operator !(Heap heap)

{

List temp = new List();

temp = heap.first;

int i = 0;

double []arr = new double[heap.amount];

while(temp!=null){

arr[i++] = temp.count;

temp = temp.next;

}

return arr[0];

}

public static Heap operator ++(Heap heap)

{

List temp = heap.first;

while(temp!=null){

temp.count += 1.0;

temp = temp.next;

}

return heap;

}

}

}

4.Выполнения программы

Было написано 2 класса: класс Program, через метод которого запускалась программа и основной класс Heap. Были написаны перегрузки операторов для нахождения максимального элемента в списке, добавление нового числа в список и увеличение всех элементов в списке на 1.

Результат запуска программы при данных вызовах в классе Program на экран выведется число 11,2, т.к. изначально максимальным было 9,2, но позже все элементы были дважды увеличены на единицу

ВЫВОДЫ

В данной лабораторной работе было изучено понятие перегрузки операторов на языке С#. Также предварительно были изучены особенности самого я зыка и его отличия от С++. Был написан класс Куча, который реализовывал три перегрузки, для работы с очередью.