# **Практическая работа № 7. Перегрузка операторов и методов**

Объём учебного времени – 2ч

Методические рекомендации

1. Цель работы: Научиться применять перегрузку операторов и методов, изучить общие принципы описания операторных методов, .
2. Перечень необходимых средств обучения:
   1. Технические средства обучения:

Компьютер Core i3 3.0, 2 Gb оперативной памяти, винчестер 250 Gb, DVD

* 1. Программное обеспечение:
* Программа ОС Windows XP/7;
* Антивирусные программы: Kaspersky AntiVirus;
* Браузер Internet Explorer;
* Интегрированная среда программирования MS Visual Studio 2017.

1. Основные теоретические положения:

В языке C# можно определять действие операторов на операнды, являющиеся объектами пользовательских классов. Этот механизм называется перегрузкой операторов и реализуется путем описания в пользовательском классе операторных методов.

Чтобы определить способ применения того или иного оператора к объектам класса, в классе описывается специальный метод, который называется операторным.

Название операторного метода состоит из ключевого слова **operator** и символа оператора, для которого определяется способ применения к объектам класса.

В языке C# перегружаться могут многие операторы, но не все.

Бинарные операторы для перегрузки :

**+, -, \*, /, %, &, |, ^, << и >>.**

Унарные операторы для перегрузки:

**+, -, !, ~, ++, --, true и false.**

Есть операторы, которые не перегружаются. Среди них:

= (присваивание),

. (оператор "точка"),

?: (тернарный оператор),

-> (оператор "стрелка", используемый при работе с указателями на структуры),

=> (оператор "стрелка", используемый при описании лямбда-выражений),

new (оператор создания нового объекта) и ряд других.

Пример\_1:

**using** System;

**using** System.Collections.Generic;

**using** System.Linq;

**using** System.Text;

**namespace** Prakt\_10

{

**class** Program

{

**// Класс с перегрузкой оператора сложения:**

**class** MyClass {

**// Целочисленное поле:**

**public** **int** number;

**// Конструктор с целочисленным аргументом:**

**public** MyClass(**int** n) {

**// Присваивание значения полю:**

number = n;

}

**// Операторный метод для перегрузки оператора сложения:**

**public** **static** **int** **operator**+(MyClass a, MyClass b) {

**// Локальная целочисленная переменная:**

**int** m = a.number + b.number;

**// Результат метода:**

**return** m;

}

}

**// Класс с главным методом:**

**static** **void** Main()

{

**// Создание объектов класса MyClass:**

MyClass A = **new** MyClass(100);

MyClass B = **new** MyClass(200);

**// Целочисленная переменная:**

**int** num;

**// Вычисление суммы объектов:**

num = A + B;

**// Отображение результата вычисления суммы объектов:**

Console.WriteLine("A + B = " + num);

**// Задержка:**

Console.ReadLine();

}

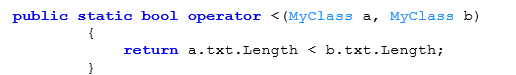
}

}

Иногда возникает необходимость создать один и тот же метод, но с разным набором параметров. И в зависимости от имеющихся параметров применять определенную версию метода. Такая возможность называется перегрузкой методов.

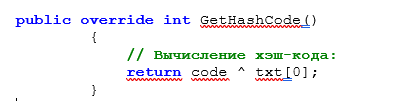
И в языке C# можно создавать в классе несколько методов с одним и тем же именем. Но при этом мы должны учитывать, что методы с одним и тем же именем должны иметь либо разное количество параметров, либо параметры разных типов.

1. Практические указания:
   1. Напишите программу, в которой есть класс с целочисленным полем и текстовым полем. Выполните перегрузку всех операторов сравнения. Сравнение на предмет "больше" или "меньше" выполняется на основе сравнения длины текстовых значений (имеются в виду текстовые поля сравниваемых объектов). При сравнении на предмет "больше или равно" или "меньше или равно" сравниваются значения целочисленных полей объектов. При сравнении на предмет "равно" или "не равно" сравниваются и целочисленные, и текстовые поля объектов. Также предложите способ переопределения методов Equals() и GetHashCode().
      1. Создайте консольное приложение.
      2. Объявите класс с именем MyClass с перегрузкой операторов сравнения == и != и переопределением методов GetHashCode() и Equals(). Поля и методы класса должны быть открытыми:
   * определите поля класса:
   * целочисленное поле code;
   * текстовое поле txt;
   * опишите конструктор класса с двумя аргументами, соответствующих типов, в теле конструктора полям code и txt присвойте значения соответствующих аргументов;
   * опишите открытый статический операторный метод, для перегрузки оператора «меньше»:



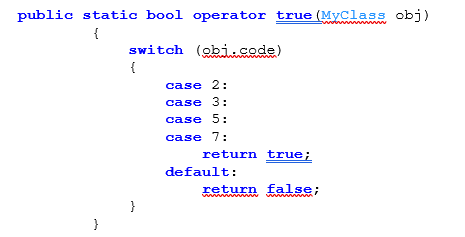
* + аналогично опишите открытый статический операторный метод, для перегрузки оператора «больше»;
  + опишите открытые статические операторные методы, для перегрузки оператора «больше или равно» и меньше или равно. По условию задачи, при сравнении на предмет «больше или равно» или «меньше или равно» сравниваются значения целочисленных полей объектов.
  + Хэш-коды используются для сравнения двух объектов на предмет равенства. Главное правило такое: если два объекта считаются одинаковыми (равными), то у них должны быть одинаковые хэш-коды. Метод GetHashCode() должен вернуть целое число. Результат вычисляется как значение выражения code^symb. Здесь использован оператор побитового исключающего или ^, а операндами являются поля code и txt[0] (для символа берется код из кодовой таблицы) объекта, из которого вызывается метод.

Для переопределения метода GetHashCode() напишите следующий текст:

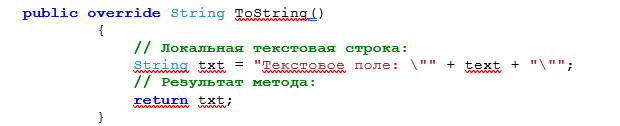


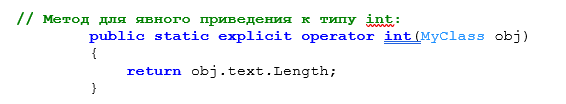
* + Для реализации и переопределения метода  Equals(), в качестве параметра метода задайте объектную переменную класса MyClass с именем obj, в теле метода объявите локальную объектную переменная t класса МуСlass и присвойте ей значение аргумента метода. Результат метода вычислите с помощью условной конструкции. Если совпадают значения полей объекта, из которого вызывается метод, и объекта, переданного аргументом методу, то метод Equals() результатом возвращает значение true, а в противном случае результат метода равен false;
  + используйте метод Equals() при реализации перегрузки оператора ==;
  + реализуйте перегрузку оператора !=.
    1. В главном модуле:
  + создайте пять объектов класса MyClass с именами A,B,C,D,E;
  + реализуйте использование операторов строгого сравнения;
  + использование операторов нестрогого сравнения;
  + сравнение объектов на предмет равенства и неравенства.

* 1. Напишите программу, в которой есть класс с целочисленным полем. Перегрузите операторы &, |, true и false так, чтобы с объектами класса можно было использовать операторы && и ||. Перегрузку следует реализовать так, чтобы объект считался "истинным", если значение его числового поля равно 2, 3, 5 или 7. Объект должен рассматриваться как "ложный", если значение его числового поля меньше 1 или больше 10.
     1. Создайте консольное приложение.
     2. Объявите класс с именем MyClass. Поля и методы класса должны быть открытыми. В классе должны быть определены:
  + целочисленное поле code;
  + опишите конструктор класса с одним аргументом целочисленного типа, в теле конструктора полю code присвойте значение соответствующего аргумента;
  + реализуйте перегрузку оператора true, используя конструкцию switch-case:



* + реализуйте перегрузку оператора false. По условию задачи объект должен рассматриваться как "ложный", если значение его числового поля меньше 1 или больше 10. Таким образом, объект будет считаться "ложным", если он не "истинный" или его числовое поле удовлетворяет указанному условию;
  + реализуйте метод operator&()  исходя из того, что если первый аргумент (объект а класса MyClass) "истинный", то результатом возвращается ссылка на второй аргумент (объект b класса MyClass). Если первый операнд не "истинный", то результатом возвращается ссылка на него;
  + реализуйте операторный метод operator|(), который  возвращает ссылку на первый операнд, если он "истинный". Если первый операнд не "истинный", то результатом метода возвращается ссылка на второй операнд.
    1. В главном модуле:
  + создайте четыре объекта класса MyClass с именами A,B,C,D;
  + выполните проверку использования логических операторов.
  1. Напишите программу, и которой есть класс с текстовым полем. Опишите в классе операторные методы для выполнения приведения типов. Необходимо определить следующие способы преобразований. При преобразовании объекта в целое число результатом возвращается количество символов в значении текстового поля. При преобразовании объекта в символ результатом является первый символ в тексте. При преобразовании числа в объект создается (и возвращается результатом соответствующего операторного метода) объект, текстовое поле которого содержит текстовую строку из символов 'А'. Количество символов в тексте определяется преобразуемым числом.
     1. Объявите класс с именем MyClass. Поля и методы класса должны быть открытыми .В классе должны быть определены:
  + текстовое поле text;
  + опишите конструктор класса с одним аргументом строкового типа, в теле конструктора полю text присвойте значение соответствующего аргумента;
  + выполните переопределение метода ToString():

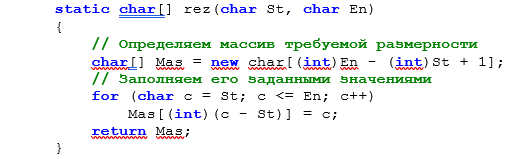


* + Для явного приведения типов, необходимо использовать ключевое слово explicit. 
  + аналогично реализуйте метод для явного приведения к типу char;
  + выполните реализацию метода для явного преобразования из типа int:



в теле метода создайте локальную переменную s, инициализированную пустым значением. Далее в цикле, поместите в s о количество символов 'A', которое совпадает со значением переменной, переданной методу в качестве аргумента. Затем эта переменная s выступает в качестве аргумента для конструктора класса MyClass, при выполнении которого создается объект t этого класса. Этот объект возвращается в качестве результата работы этого метода.

* + 1. В главном методе создайте необходимые объекты и выполните проверку всех методов.
  1. Напишите программу с перегруженным статическим методом. Если методу передается два целых числа, то результатом возвращается ссылка на целочисленный массив, состоящий из натуральных чисел, а первое и последнее число в массиве определяется аргументами метода. Например, если передать аргументами числа 2 и 4, то результатом будет массив из чисел 2, 3 и 4. Если методу передаются два символьных значения, то результатом возвращается ссылка на массив, состоящий из последовательности символов, а первый и последний символы определяются аргументами метода. Например, если передать аргументами методу символы 'В' и 'D', то в результате получим массив из символов 'В', 'С' и 'D'.
  + напишите статический метод для символов:



* + самостоятельно напишите перегруженный метод для чисел;
  + напишите метод для отображения одномерного целочисленного массива;
  + напишите метод для отображения одномерного символьного массива.
    1. В главном модуле выполните проверку всех методов.
  1. Задания для индивидуального выполнения:

1. Контрольные вопросы
2. Перегрузка операторов это…
3. Что такое операторный метод?
4. Перечислите перегружаемые операторы?
5. Перегрузка методов это..?
6. Список рекомендуемой литературы:

**Основная литература:**

1. Немцова Т.И. Програмиирование на языке выского уровня. Программирование на языке Object Pascal: учеб. пособие /Т.И. Немцова, С.Ю. Голова, И.В. Абрамова. – М.: ФОРУМ: ИНФРА – М, 2012. – 496 с.
2. Овечкин Г.В. Компьютерное моделирование: учебник для студ. учреждений сред. проф. образования / Г.В. Овечкин.- М.: Издательский центр «Академия», 2015.-224с.
3. Немцова, Т. И.  Программирование на языке высокого уровня. Программирование на языке С++ : учеб. пособие / Т. И. Немцова, С. Ю. Голова, А. И. Терентьев. - М. : ФОРУМ : ИНФРА-М, 2012. - 512 с. : ил. + CD.
4. Гуриков, С. Р. Введение в программирование на языке Visual C# : учеб. пособие / С. Р. Гуриков. - М. : ФОРУМ : ИНФРА-М, 2013. - 448 с.

**Дополнительная литература:**

1. Рао Сиддхартха Освой самостоятельно С++ за 21 день, 7 изд.: Пер с англ.-М.: ООО «И.Д.Вильямс», 2013 – 688 с.: ил. – Парал.тит.англ.
2. Голицына О.Л. Программное обеспечение: учеб. пособие для среднего профессионального образования-М.:ФОРУМ;ИНФРА-М,2006.-432 с.
3. Виллемер А. Программирование на С++/А. Виллемер;[пер. с нем. М.А.Райтман].-М.:Эксмо,2013.-528с.+CD.-(Мировой компьютерный бестселлер).
4. Культин Н.Б. Microsoft Visual C++ в задачах и примерах.- СПб.: БХВ-Петербург,2010.-272 с.:ил.+CD-ROM.
5. Партыка Т.Л. Операционные системы, среды и оболочки: учеб. пособие для студ. учреждений сред. проф. Образования/ Т.Л. Партыка, И.И. Попов. – 3-е изд., перераб. и доп. – М.: ФОРУМ, 2010. – 543 с.
6. Окулов С.М. Основы программирования: учебное пособие.- М.:БИНОМ. Лаборатория знаний, 2010.- 440 с.
7. Голицына О.Л. Основы алгоритмизации и программирования: учеб. пособие для сред. проф. образования.- М: ФОРУМ; ИНФРА-М,2005.-432 с.
8. Рихтер Дж. Программирование приложений для Microsoft Windows /Пер. с англ. – M.: Microsoft Press, 2003. – C.48-313.