

Практическая работа 23

Перегрузка операторов и методов

Цель занятия: Получить практические навыки создания перегруженных операторов и методов в классах

Перечень оборудования и программного обеспечения

Персональный компьютер
Microsoft Office (Word, Visio)
Microsoft Visual Studio

Краткие теоретические сведения

Полиморфизм - обеспечивает возможность задания различных реализаций некоторого единого по названию метода для классов различных уровней иерархий. Язык программирования поддерживает полиморфизм, если классы с одинаковой спецификацией могут иметь различную реализацию — например, реализация класса может быть изменена в процессе наследования

В C# допускается совместное использование одного и того же имени двумя или более методами одного и того же класса, при условии, что их параметры объявляются по-разному.

Существование в классе методов с одним и тем же именем называется перегрузкой, а сами одноименные методы называются перегруженными. Перегрузка методов полезна, когда требуется решать подобные задачи с разным набором аргументов. Перегрузка методов относится к одному из способов реализации полиморфизма в C#.

Перегрузка характерна и для знаков операций. В зависимости от типов аргументов, один и тот же знак может выполнять фактически разные операции. Классическим примером является знак операции сложения «+», который играет роль операции сложения не только для арифметических данных разных типов, но и выполняет конкатенацию строк.

Перегрузка требует уточнения семантики вызова метода. Когда встречается вызов неперегруженного метода, то имя метода в вызове однозначно определяет, тело какого метода должно выполняться в точке вызова. Когда же метод перегружен, то знания имени недостаточно - оно не уникально. Уникальной характеристикой перегруженных методов является их сигнатура. Перегруженные методы, имея одинаковое имя, должны отличаться либо числом аргументов, либо их типами, либо ключевыми словами (отметим: с точки зрения сигнатуры, ключевые слова **ref** и **out** не отличаются). Уникальность сигнатуры позволяет вызвать требуемый перегруженный метод.

Перегрузка широко используется при построении библиотеки FCL. Например, класс **Convert**, у которого 16 методов с разными именами, зависящими от целевого типа преобразования. Каждый из этих 16 методов перегружен, и в свою очередь, имеет 16 реализаций в зависимости от типа источника.

Пример.

Метод будет добавлять аргументы друг к другу и выводить результат:

```
public static void AddAndDisplay(char a, char b)
{
    Console.WriteLine(a.ToString() + b.ToString());
}
public static void AddAndDisplay( bool a, bool b)
{
    Console.WriteLine(a || b);
}

static void Main(string[] args)
{
    AddAndDisplay(5, 8); // 13
    AddAndDisplay('C', '#'); // "C#"
    AddAndDisplay(true, false); // "C#"
    Console.ReadKey();
}
```

При вызове метода, компилятор сам решает, какой вариант метода `AddAndDisplay` необходимо вызвать.

Совершенно недостаточно, чтобы методы отличались только типами возвращаемых значений. Они могут также отличаться числом своих параметров. Когда вызывается перегружаемый метод, то выполняется тот его вариант, параметры которого соответствуют (по типу и числу) передаваемым аргументам.

Пример:

```
class GreetDemo
{
    static void Greet()
    {
        Console.WriteLine("Hello");
    }
    static void Greet(string Name)
    {
        Console.WriteLine("Hello " + Name)
    }
    static void Main( )
    {
        Greet( );
        Greet("Alex");
    }
}
```

}

Задания

- 1 Изучить теоретические сведения и задание к работе
- 2 Используя задание практической работы 22, создать конструкторы и методы класса
- 3 В соответствии с вариантом задания составить отлаженную программу.

Порядок выполнения работы

Изучить теоретические сведения и задание к работе
В соответствии с вариантом задания составить отлаженную программу.

Содержание отчета

- 1 Название работы
- 2 Цель работы
- 3 Технические средства обучения
- 4 Задания (условия задач)
- 5 Порядок выполнения работы
- 6 Листинг отлаженной программы в соответствии с вариантом задания
- 7 Ответы на контрольные вопросы
- 8 Вывод

Варианты заданий

Создать в классе методы, различающиеся типом возвращаемых значений и типами аргументов, содержащие перегрузку метода:

1. Сумма (одномерные массивы, логические значения, целые значения)
2. Разность (массивы, строки, целые значения)
3. Произведение (матрицы, логические значения, целые значения)
4. Пересечение (массивы, строки, прямоугольники)
5. Объединение (массивы, строки, прямоугольники)
6. Отрицание (логические значения, строки)
7. Сумма (вектора, логические значения, вещественные значения)
8. Разность (вектора, окружности, вещественные значения)
9. Произведение (вектора, логические значения, вещественные значения)
10. Пересечение (окружности, прямые)

- 11.Объединение (окружности, смежные углы)
- 12.Эквивалентность (массивы, логические значения, строки)
- 13.Сумма (вектора, логические значения, вещественные значения)
- 14.Произведение (вектора, логические значения, вещественные значения)
- 15.Пересечение (окружности, прямые)
- 16.Разность (массивы, строки, целые значения)
- 17.Произведение (массивы, логические значения, целые значения)
- 18.Разность (вектора, окружности, вещественные значения)
- 19.Отрицание (логические значения, строки)
- 20.Пересечение (окружности, прямые)
- 21.Сумма (одномерные массивы, логические значения, целые значения)
- 22.Разность (массивы, строки, целые значения)
- 23.Объединение (окружности, смежные углы)

Контрольные вопросы

1. Что такое полиморфизм?
2. Для чего предназначена перегрузка?
3. К каким элементам класса применяется перегрузка?
4. Какие характеристики оказывают влияние на идентификацию метода и могут быть перегружены?
5. Можно ли перегружать встроенные операции C#?
6. Общая форма перегрузки оператора.
7. Чем отличаются перегрузки унарного и бинарного операторов?

Используемая литература

- 1 Гниденко, И. Г. Технология разработки программного обеспечения: учеб. пособие для СПО / И. Г. Гниденко, Ф. Ф. Павлов, Д. Ю. Федоров. — М.: Издательство Юрайт, 2017.
- 2 Шарп Джон Ш26 Microsoft Visual C#. Подробное руководство. 8-е изд. — СПб.: Питер, 2017.
- 3 Васильев А.Н. Программирование на C# для начинающих. Основные сведения. – Москва: Эксмо, 2018.
- 4 Васильев А.Н. Программирование на C# для начинающих. Особенности языка. – Москва: Эксмо, 2019.
- 5 <http://msdn.microsoft.com/ru-ru/library/67ef8sbd.aspx>.