**1. Перечислите свойства внутренних и вложенных классов?**

вложенный класс должен быть объявлен с любым модификатором доступа кроме private.

То есть, вызов конструктора экземпляра через ключевое слово new невозможен.

**2. Что такое статический класс и какие у него свойства?**

Означает, что этот **класс** является набором только **статических** сущностей (полей, методов, свойств и т. ... п.), и, создавать объекты такого **класса** нельзя

**3. Каково назначение перегрузки операторов?**

**Перегрузка** - один из способов реализации [полиморфизма](https://ru.wikipedia.org/wiki/%D0%9F%D0%BE%D0%BB%D0%B8%D0%BC%D0%BE%D1%80%D1%84%D0%B8%D0%B7%D0%BC_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)), заключающийся в возможности одновременного существования в одной [области видимости](https://ru.wikipedia.org/wiki/%D0%9E%D0%B1%D0%BB%D0%B0%D1%81%D1%82%D1%8C_%D0%B2%D0%B8%D0%B4%D0%B8%D0%BC%D0%BE%D1%81%D1%82%D0%B8) нескольких различных вариантов применения оператора, имеющих одно и то же имя, но различающихся типами параметров, к которым они применяются

**4. Как используется ключевое слово operator?**

Определяет  операторный метод, который, определяет действие оператора относительно своего класса

**5. Какие операции нельзя перегружать в C#?**

* ?: (тернарный оператор);
* :: (доступ к вложенным именам);
* . (доступ к полям);
* . \* (доступ к полям по указателю);
* sizeof , typeid и операторы каста.
* [], ->, (), += -= \*= /= и тд.

**6. Можно ли перегрузкой отменить очередность выполнения операции?**

Да

**7. Истинно ли следующее утверждение: операция >= может быть перегружена.**

Да

**8. Сколько аргументов требуется для определения перегруженной унарной операции?**

1

**9. Можно ли перегружать операцию []?**

*можно перегружать* только *как* метод (НЕТ)

**10. Можно ли перегружать операцию ->?**

*можно перегружать* только *как* метод (НЕТ)

**11. Приведите пример оператора приведения типа**

public static bool operator >(IntArray arr1, IntArray arr2)

{

if (arr1.Count > arr2.Count)

{

return true;

}

else return false;

}

**12. Что такое метод расширения? Как и где его можно использовать?**

Методы расширения (extension methods) позволяют добавлять новые методы в уже существующие типы без создания нового производного класса.

**13. Пусть дан фрагмент кода определения оператора преобразования типа. Определить форму преобразования.**

public static implicit operator Point2D(Point3D a) {/\* код\*/;}

преобразование из 3d в 2d

**14. Выберите верное утверждение. Метод расширения может:**

1) получать доступ к public членам расширяемого класса

2) получать доступ к protected членам расширяемого класса

3) получать доступ к internal членам расширяемого класса

4) быть объявлен в любом классе

5) быть без параметров

**15. Выберите из списка неверное правило перегрузки операторов для C#.**

1) префиксные операции ++ и – – перегружаются парами

2) операции сравнения перегружаются парами: == и != ; < и >;<= и >=

3) перегруженные операции обязаны возвращать значения

#### 4) должны объявляться как protected

5) true и false можно перегружать