**1. Назовите принципы ООП. Поясните каждый из них.**

* Инкапсуляция – связывание вместе кода и данных и защита их от неправильного использования.
* Полиморфизм – способность программы одинаково использовать объекты с одинаковым интерфейсом, независимо от их методов; способность вызывать метод потомка через экземпляр предка.
* Абстракция – разделение интерфейса и реализации

**2. Назовите класс .NET, от которого наследуются все классы.**Object

**3. Охарактеризуйте открытые методы System.Object.**

ToString (возвращает полное имя), GetHashCode, Equals

**4. Охарактеризуйте закрытые методы System.Object.***Finalize* - Вызывается, когда сборщик мусора определяет, что объект является мусором, но до возвращения занятой объектом памяти в кучу.

*GetType –* возвращает тип

*Clone –* создает новый экземпляр, поля которого идентичны исходному.

**5. Приведите пример определения класса.**

public class Person{}

**6. Какие ключевые слова можно использовать при определении класса?**Class

**7. В чем отличие между объектом и классом?**Объект – экземпляр класса

**8. Что такое конструктор? Когда вызывается конструктор?**Специальные методы, которые выполняют инициализацию объектов. Вызываются при создании объекта.

**9. Перечислите свойства конструктора?**

* Не возвращает значений
* Имя совпадает с именем класса
* Не наследуется
* Один класс может иметь несколько конструкторов
* Один конструктор может вызывать другой
* Если в классе не определены конструкторы, то компилятор вызывает конструктор по умолчанию(без параметров)
* нельзя применять модификаторы virtual, new, override, sealed и abstract
* для статических классов (seald и abstract) компилятор не создает конструктор по умолчанию

**10. Что такое деструктор (destructor) ?**деструктор вызывается перед окончательным уничтожением объекта системой сборки мусора. Иногда может вообще не вызваться.

**11. Что такое this?**Ключевое слово, обеспечивающее доступ к текущему экземпляру класса.

**12. Что будет выведено в результате выполнения**5 5

7 7

**13. Какие спецификаторы доступа для класса и методов класса**

**существуют в C#?**

* **public:** публичный, общедоступный класс или член класса. Такой член класса доступен из любого места в коде, а также из других программ и сборок.
* **private:** закрытый класс или член класса. Представляет полную противоположность модификатору public. Такой закрытый класс или член класса доступен только из кода в том же классе или контексте.
* **protected:** такой член класса доступен из любого места в текущем классе или в производных классах. При этом производные классы могут располагаться в других сборках.
* **internal:** класс и члены класса с подобным модификатором доступны из любого места кода в той же сборке, однако он недоступен для других программ и сборок (как в случае с модификатором public).
* **protected internal**: совмещает функционал двух модификаторов. Классы и члены класса с таким модификатором доступны из текущей сборки и из производных классов.
* **private protected:** такой член класса доступен из любого места в текущем классе или в производных классах, которые определены в той же сборке.

**14. Опишите модификатор protected internal.**

**protected internal:** совмещает функционал двух модификаторов. Классы и члены класса с таким модификатором доступны из текущей сборки и из производных классов.

**15. Зачем и как используются ref и out параметры функции?**Ref и Out – параметры-ссылки. Ref передает значение по ссылке, out работает как дополнительный return.

**16. Приведите пример необязательных и именованных параметров метода.**

* Необязательный static void RedrawButton(int color , int type = 2 , int size = 4)
* Именованный RedrawButton(243,size:45);

**17. Приведите пример полей класса – статические, константные, только для чтения.**static string type;

private const string airport = "Минск-1";

private readonly int planeID;

**18. Приведите пример определения свойств класса. Как свойства связаны с  
инкапсуляцией?**Свойства определяют методы записи и чтения. Свойства применяются к закрытым полям класса и служат для организации доступа к ним(представляют инкапсуляцию).

**19. Назовите явное имя параметра, передаваемого в метод set свойства**

**класса?**value

**20. Что такое автоматические свойства?**Имеют сокращенную запись, и компилятор автоматически реализует нужные методы.

**21. Что такое индексаторы класса? Какие ограничения существуют на  
индексатор?**Индексаторы обеспечивают возможность доступа к элементам класса по их порядковому номеру.

Ограничения:

* Значение, выдаваемое индексатором, нельзя передавать методом в качестве ref или out параметра
* Индексатор не может быть static

**22. Что такое перегруженный метод?**Метод, дублирующий уже существующий, но имеющий другой набор пораметров.

**23. Что такое partial класс и какие его преимущества?**Частичный класс, т.е. может существовать несколько файлов с определением одного и того же класса, и при компиляции они будут объединены. Используется для управления версиями, разделения файла на логические части, использование шаблонов.

**24. Что такое анонимный тип в C#?**Тип, задаваемый ключевым словом var, позволяет создать объект с некоторым набором свойств без определения класса.

**25. Для чего делают статические классы?**

* Для хранения связанных друг с другом статических методов
* Для создания метода расширения

**26. В чем отличие статического поля от экземплярного?**Для использования статических полей класса не обязательно создавать экземпляр.

**27. Поясните работу статических конструкторов.**Используются для инициализации статических данных или выполняют действие, которое нужно выполнить только один раз. Вызываются при компиляции первыми и только один раз.

**28. Какая разница между поверхностным (shallow) и глубоким (deep)  
копированием?**При неглубоком копируются только ссылки на объекты, а не сами объекты.

**29. В чем разница между равенством и тождеством объектов?**Равенство, если объекты имеют одинаковые значения, тождественность – что они ссылаются на один объект.

**30. Что такое частичные классы и частичные методы?**Классы и методы, которые логически разделены в файле.

**31. Что будет выведено на консоль результате выполнения следующего кода:**

System.Int32 **32. Что будет выведено на консоль результате выполнения следующего кода:** 2 1  **33. Пусть задан следующий класс.  
Какой из конструкторов задан неверно?** 2 **34. Пусть задан следующий класс.  
Сколько аргументов может быть задано при вызове конструктора  
данного класса?** 2  **35. Почему не удается создать объект класса A?**

Неправильно описан конструктор 1,2,3, а для четвертого не хватает параметров **36. Что будет выведено в консоль при выполнении данной программы?**

A static (статический конструктор вызывается ПЕРВЫМ и один раз)

A **37. Какая строка приведенного далее класса вызовет ошибку компиляции?** 3