1. Назовите принципы ООП. Поясните каждый из них.

Инкапсуляция – сокрытие кода и данных, позволяет исключить вмешательство из вне.

Наследование – можно строить новые определения классов на основе предыдущих

Полиморфизм – он обозначает способность языка трактовать связанные объекты в сходной манере.

1. Назовите класс .NET, от которого наследуются все классы.

Object.

1. Охарактеризуйте открытые методы System.Object.

ToString(), GetHashCode(), GetType(), Equals()

1. Охарактеризуйте закрытые методы System.Object.

Void Finalize(), Object MemberwiseClone

1. Приведите пример определения класса.

public class Human

{

public string firstname;

public string secondname;

public double age;

}

1. Какие ключевые слова можно использовать при определении класса?

class

1. В чем отличие между объектом и классом?

Объект является экземпляром класса.

1. Что такое конструктор? Когда вызывается конструктор?

Конструкторы вызываются при создании нового объекта класса. Конструкторы выполняют инициализацию объекта.

1. Перечислите свойства конструктора?
2. Что такое деструктор (destructor) ?

В деструктор вкладывается логика освобождения неуправляемых ресурсов. Задаётся при помощи знака ~

1. Что такое this?

Обеспечивает доступ к текущему экземпляру класса.

1. Что будет выведено в результате выполнения

5 5

7 7

1. Какие спецификаторы доступа для класса и методов класса существуют в

C#?

Public, private, protected, internal, protected internal, private protected

1. Опишите модификатор protected internal.

Классы и члены класса с таким модификатором доступны из текущей сборки и из производных классов.

1. Зачем и как используются ref и out параметры функции?

ref используется для передачи не копии значения переменной, а ссылки на переменную. Out работает как и return, позволяет получить значение из метода, даже если функция является типа void.

1. Приведите пример необязательных и именованных параметров метода.

Public static void primer1(int x, int y =5, int z) {}y – необязательный параметр.

Public static void primer2(int x, int y, int z){}

public static void primer2.1(){

primer2(y:3,x:2,z:4);

}

17. Приведите пример полей класса – статические, константные, только для

чтения.

static int prim;

const int prim2;

readonly string prim3;

18. Приведите пример определения свойств класса. Как свойства связаны с

инкапуляцией?

class Person

{

    private string name;

    public string Name

    {

        get

        {

            return name;

        }

        set

        {

            name = value;

        }

    }

}

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

Value

20. Что такое автоматические свойства?

Это сокращенная запись св-в - {get; set;}

21. Что такое индексаторы класса? Какие ограничения существуют на

индексатор?

Индексаторы позволяют индексировать объекты и обращаться к ним по индексу.

22. Что такое перегруженный метод?

Позволяет создать метод с одним и тем же названием но с разными аргументами.

23. Что такое partial класс и какие его преимущества?

Позволяет записать определения класса, в разные файлы, которые будут скомпилированы в один класс.

24. Что такое анонимный тип в C#?

Тип который не принадлежит конкретному классу

например : Var prim = new {имена и их значения}

25. Для чего делают статические классы?

Статический класс может использоваться как обычный контейнер для наборов методов, работающих на входных параметрах, и не должен возвращать или устанавливать каких-либо внутренних полей экземпляра.

26. В чем отличие статического поля от экземплярного?

Доступ к статическому полю в отличие от экземплярного производится через имя класса, а не объект.

27. Поясните работу статических конструкторов.

Статические конструкторы не должны иметь модификатор доступа и не принимают параметров. В стат. Конструкторах нельзя использовать ключевое слово this. Статические конструкторы нельзя вызвать вручную, они вызываются сами при первом создании объекта.

Статические конструкторы используются для инициализации статических данных, либо для однократных действий.

28. Какая разница между поверхностным (shallow) и глубоким (deep)

копированием?

При поверхностном копировании копируются значения полей класса, включая значения любых указателей или ссылок. При глубоком копировании мы копируем значения полей не только на первом уровне, но и копируем все значения.

29. В чем разница между равенством и тождеством объектов?

30. Что такое частичные классы и частичные методы?

Частичный класс – класс, который определён в нескольких файлах, но будет скомпилирован, как один. Определение частичного метода может находится в одном классе, а реализация -- в другом. Задаются они ключевым словом partial.

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

System.Int32

32. Что будет выведено на консоль результате выполнения следующего кода:

a =2, b=1

33. Пусть задан следующий класс.

Какой из конструкторов задан неверно?

2ой

34. Пусть задан следующий класс.

Сколько аргументов может быть задано при вызове конструктора

данного класса?

Минимум – 0, максимум – 2;

35. Почему не удается создать объект класса A?

36. Что будет выведено в консоль при выполнении данной программы?

A static

A

37. Какая строка приведенного далее класса вызовет ошибку компиляции?

b = 30; //3