1. Назовите принципы ООП. Поясните каждый из них.

Абстракция данных: подробности внутренней логики скрыты от конечного пользователя. Пользователю не нужно знать, как работают те или иные классы и методы, чтоб их использовать.

Наследование: делает возможным повторное использование кода — если какой-то класс уже имеет какую-то логику и функции, нам не нужно переписывать всё это заново для создания нового класса, мы можем просто включить старый класс в новый, целиком.

Полиморфизм: это свойство одних и тех же объектов и методов принимать разные формы.

Инкапсуляция: ограничивает клиентов от просмотра своего внутреннего представления, в котором реализовано поведение абстракции.

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

System.Object

3. Охарактеризуйте открытые методы System.Object.

Equals (сравнение между объектами), Finalize(), GetHashCode(создаёт хеш-код), ToString(создаёт строку с описанием объекта)

4. Охарактеризуйте закрытые методы System.Object.

GetType(позволяет получить тип объекта)

5. Приведите пример определения класса.

6. Какие ключевые слова можно использовать при определении класса?

public

private (доступен в рамках своего класса или структуры)

protected (доступ к элементу может быть получен из соответствующего класса или его экземпляров)

internal (доступен только в рамках своей сборки)

abstract (элемент имеет неполную реализацию)

sealed (запрещает наследовать от класса)

7. В чем отличие между объектом и классом?

Класс – некоторая структура, что описывает характеристики и поведение, объект – экземпляр такого класса, что-то конкретное.

8. Что такое конструктор? Когда вызывается конструктор?

Специальный метод, который используется для инициализации объекта. Вызывается во время создания объекта.

9. Перечислите свойства конструктора?

– не возвращает значение

– невозможно использовать оператор return

– не могут быть виртуальными

10. Что такое деструктор (destructor)?

Конструктор создает, деструктор уничтожает элементы класса.

11. Что такое this?

Ключевое слово, которое ссылается на текущий экземпляр класса.

12. Что будет выведено в результате выполнения?

5 5

7 7

13. Какие спецификаторы доступа для класса и методов класса существуют в C#? (ответ в шестом вопросе)

14. Опишите модификатор protected internal.

Совмещает свойства двух идентификаторов доступа.

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

Используются для передачи методу параметров по адресу. Сообщают компилятору, что адреса используемых переменных совпадают с адресами переменных, передаваемых в качестве параметров.

16. Приведите пример необязательных и именованных параметров метода.

Используем именованные параметры, если, к примеру, не помним их порядок расположения, но помним имена. Необязательные прописываются в конце.

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

Статические поля просто хранят какое-то общее состояние, константные – значение такого поля нельзя изменить, только для чтения – для объявления используется ключевое слово readonly.

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

инкапсуляцией?

Доступ к ним можно получить только через экземпляр класса.

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

value

20. Что такое автоматические свойства?

get и set в фигурных скобках

21. Что такое индексаторы класса? Какие ограничения существуют на

индексатор?

Позволяют обращаться с объектами как с массивами дополнительно с помощью ключевого слова this.

22. Что такое перегруженный метод?

Иногда появляется необходимость создать один метод, но с разным набором параметров. В зависимости от того, какие параметры были переданы, будет вызван определенный из методов.

23. Что такое partial класс и какие его преимущества?

Позволяет разбить кусок кода на два файла.

24. Что такое анонимный тип в C#?

Позволяют создать класс на лету во время компиляции, методы внутри определять нельзя.

25. Для чего делают статические классы?

Экземпляры статического класса создавать нельзя. Пример статического класса – Math. Создаются в основном как контейнеры для набора методов, работающих на входных параметрах.

26. В чем отличие статического поля от экземплярного?

27. Поясните работу статических конструкторов.

Используется для инициализации статических данных и вызывается чтобы выполнить действие всего один раз перед созданием первого объекта класса.

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

копированием?

При поверхностном мы копируем указатели и ссылки на объект, при глубоком – все значения.

29. В чем разница между равенством и тождеством объектов?

Равные объекты располагаются в разных участках памяти и все их поля равны, тождественные переменные ссылаются на один и тот же объект на одном участке памяти, поэтому и называются тождественными.

30. Что такое частичные классы и частичные методы?

Могут располагаться в разных файлах, не имеют модификаторов доступа и не возвращают значений.

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

System.Int32