1.Что такое наследование?

Ответ: Наследование в объектно-ориентированном программировании (ООП) позволяет новому классу (подклассу) использовать свойства и методы существующего класса (базового класса). Подкласс может расширять или изменять функциональность базового класса.

2.Что такое массив?

Ответ: Массив в C# - это упорядоченная коллекция элементов одного типа. Каждый элемент в массиве имеет индекс, который используется для доступа к этому элементу.

3.Что такое ООП?

Ответ: ООП (Объектно-Ориентированное Программирование) - это подход к программированию, основанный на использовании объектов, которые объединяют данные и методы, работающие с этими данными, в единый компонент.

4.Что такое индекс массива?

Ответ: Индекс массива - это целочисленное значение, которое используется для доступа к элементу массива. Индексация в массивах начинается с 0.

5.Какого типа данных условие цикла?

Ответ: Условие цикла должно быть логического типа данных, т.е., иметь тип bool.

6.Что такое размерность массива?

Ответ: Размерность массива определяет количество индексов, необходимых для доступа к конкретному элементу массива. Например, одномерный массив имеет одну размерность, двумерный - две, и так далее.

7.Какую библиотеку нужно подключить для работы с файлами?

Ответ: Для работы с файлами в C# используется пространство имен System.IO.

8.Определение метода Concat.

Ответ: Метод Concat в C# объединяет две строки в одну. Например:

string result = string.Concat("Hello, ", "World!");

9.Какой метод может вернуть количество символов в строке?

Ответ: Метод Length возвращает количество символов в строке. Например:

string str = "Hello";

int length = str.Length; // length будет равен 5

10.Что такое двумерный массив?

Ответ: Двумерный массив - это массив массивов. Он имеет две размерности, и каждый элемент является массивом.

11.Что такое цикл?

Ответ: Цикл в программировании - это управляющая конструкция, позволяющая выполнить набор инструкций несколько раз в зависимости от условия.

12.Что такое алгоритм?

Ответ: Алгоритм - это последовательность шагов или правил, предназначенных для решения конкретной задачи или выполнения определенной операции.

13.Назвать основные парадигмы ООП.

Ответ: Основные парадигмы ООП включают в себя инкапсуляцию, наследование и полиморфизм.

14.Наследуются ли переменные с модификатором private?

Ответ: Да, переменные с модификатором private наследуются в базовом классе, но они не являются видимыми в производных классах.

15.Кому доступны переменные с модификатором protected на уровне класса?

Ответ: Переменные с модификатором protected доступны как самому классу, так и его производным классам.

16.Назовите класс .NET, от которого наследуются все классы?

Ответ: Все классы в .NET наследуются от класса System.Object.

17.Что такое void? Где используется?

Ответ: void - это тип данных, который указывает на отсутствие возвращаемого значения. Он используется в сигнатуре метода, чтобы указать, что метод не возвращает результат.

18.Для чего служат try, catch, finally? В каком случае может не выполниться блок finally?

Ответ: Конструкция try используется для блока кода, в котором может возникнуть исключение. catch обрабатывает исключение, а finally содержит код, который выполняется всегда, независимо от того, произошло исключение или нет. Блок finally может не выполниться в случае, если происходит фатальная ошибка, такая как выход из приложения.

19.Что такое класс?

Ответ: Класс в C# - это шаблон для создания объектов. Он определяет состояние (переменные) и поведение (методы) объекта.

20.Какие модификаторы доступа существуют? Чем они отличаются?

Ответ: Существуют модификаторы доступа public, private, protected, internal и protected internal. Они определяют видимость членов класса в других частях программы.

21.Что такое объект класса?

Ответ: Объект класса - это конкретный экземпляр класса, созданный в памяти, который может использоваться для доступа к членам класса.

22.Какие типы данных есть в C#? На какие группы их можно поделить?

Ответ: Типы данных в C# можно разделить на примитивные (целочисленные, вещественные, символьные, логические) и составные (классы, массивы, интерфейсы и др.).

23.Что такое структура данных?

Ответ: Структура данных - это способ организации и хранения данных для эффективного использования и выполнения операций над ними.

24.Какие структуры данных вы знаете? И чем они отличаются друг от друга?

Ответ: Примеры структур данных включают массивы, связанные списки, стеки, очереди, деревья и графы. Они отличаются по способу организации и доступа к данным.

25.Что такое JSON?

Ответ: JSON (JavaScript Object Notation) - это легко читаемый формат обмена данными, основанный на синтаксисе JavaScript. Он представляет собой текстовый формат, используемый для передачи структурированных данных между программами.

26.В чем различие между многопоточностью и асинхронностью?

Ответ: Многопоточность - это концепция, при которой несколько потоков исполнения выполняются параллельно, что позволяет эффективнее использовать ресурсы процессора. Асинхронность - это подход, при котором операции могут выполняться асинхронно, не блокируя основной поток выполнения.

27.Как и с помощью чего реализуется полиморфизм?

Ответ: Полиморфизм в C# реализуется с помощью виртуальных методов, абстрактных классов, интерфейсов и перегрузки методов. Это позволяет объектам одного типа использовать интерфейсы объектов другого типа.

28.Назовите в чем отличие интерфейса от абстрактного класса?

Ответ: Основное отличие между интерфейсом и абстрактным классом заключается в том, что интерфейс содержит только сигнатуры методов, но не предоставляет реализации, тогда как абстрактный класс может содержать как сигнатуры методов, так и их реализации. Класс может реализовывать несколько интерфейсов, но может наследовать только один абстрактный класс.