**1.Что такое .Net Framework и из чего он состоит?**

Microsoft.NET (.NET Framework) – программная платформа. Содержит следующие основные компоненты: CLR,Компилятор JIT(Just in Time),FCL (Framework Class Library) – реализация CLI BCL компанией Microsoft. Можно рассматривать, как API CRL. обеспечивает совместное использование разных языков программирования, а также безопасность, переносимость программ и общую модель программирования для платформы Windows

**1. Что такое .Net Framework и из чего он состоит?**

обширный набор написанных фрагментов кода (созданный и поддерживающийся Microsoft),

который программисты могут использовать, чтобы быстрее писать программы.

К примеру, .NET Framework берёт на себя все операции по отрисовке окон на экране.

Программисту остаётся только вставить текст, продумать меню программы, задать поведение кнопок,

когда на них нажимает пользователь и т.д.

Но .NET Framework — намного больше, чем просто набор дополнительного кода.

Он включает в себя инструменты, призванные сократить время разработки и дополнительные API,

которые программисты могут использовать для простого взаимодействия с такими сервисами как Windows Store.

Вместо того чтобы вручную писать весь необходимый код для поддержки универсальной платформы Windows,

можно воспользоваться .NET Framework

**2. Что такое CLR, FCL/BCL, CLI, IL?**

CLI (Common Language Infrastructure) – спецификация общеязыковой инфраструктуры. Определяет архитектуру исполнительной системы и набор представляемых сервисов.

CLR (Common Language Runtime) – общеязыковая среда исполнения, виртуальная машина на которой исполняются все приложения, работающие в среде .NET.

MSIL(Microsoft IL) – реализация CLI CIL компанией Microsoft.

FCL (Framework Class Library) – реализация CLI BCL компанией Microsoft. Можно рассматривать, как API CRL. обеспечивает совместное использование разных языков программирования, а также безопасность, переносимость программ и общую модель программирования для платформы Windows.

CLR (Common Language Runtime) – Среда Времени Выполнения или Виртуальная Машина. Обеспечивает выполнение сборки (управление памятью, загрузка сборок, безопасность, обработка исключений, синхронизация)

FCL (.NET Framework Class Library) – соответствующая CLS спецификации объектно-ориентированная библиотека классов, интерфейсов и системы типов (типов-значений)

**3. Пояснить работу JIT-компилятора?**

1) CLR ищет типы данных и загружает во внутренние структуры

2) Для каждого метода CLR заносит адрес внутренней CLR функции JITCompiler

3) JITCompiler ищет в метаданных соответствующей сборки IL-код вызываемого метода, проверяет и компилирует IL-код в машинные команды

4) Они хранятся в динамически выделенном блоке памяти.

5) JITCompiler заменяет адрес вызываемого метода адресом блока памяти, содержащего готовые машинные команды

6) JITCompiler передает управление коду в этом блоке памяти.

**3. Пояснить работу JIT-компилятора?**

JIT читает байт-код из некоторых секторов (редко сразу из всех) и компилирует их в машинный код.

Этим сектором может быть файл, функция или любой фрагмент кода. Единожды скомпилированный код

может кэшироваться и в дальнейшем повторно использоваться без перекомпиляции.

**4. Что такое CTS (Common Type System)?**

CTS (Common Type Systems)- спецификацию типов, которые должны поддерживаться всеми языками ориентированными на CLR. Microsoft выпустил несколько компиляторов соответствующих этой спецификации: С++/CLI (C++ с управляемыми расширениями), С#, VB .NET, JScript.

**5. Какие аспекты поведения определяет тип System.Object?**

Тип стоит во главе иерархии классов. Все типы наследуются от Object.

**6. Что находится в mscorlib dll?**

MsCoRLib - это аббревиатура от **M**icro**s**oft **Co**mmon Language **R**untime **Lib**rary

Mscorlib.ni.dll - это файл DLL в Windows. DLL - это аббревиатура от **D**ynamic **L**ink **L**ibrary (динамически подключаемая библиотека). DLL файлы необходимы программам или расширениям веб-браузера, потому что они содержат программный код, данные и ресурсы. Бесплатный форум с информацией о файлах может помочь вам разобраться является ли mscorlib.ni.dll вирусом, трояном, программой-шпионом, рекламой, которую вы можете удалить, или файл принадлежит системе Windows или приложению, которому можно доверять.

**7. Что такое «сборка»? Из чего состоит сборка .NET?**

1) это абстрактное понятие, для логической группировки одного или нескольких управляемых модулей или файлов ресурсов. ►2) дискретная единица многократно используемого кода внутри CLR

**8. Какие виды сборок существуют?**

с нестрогими именами (weakly named assemblies)

со строгими именами (strongly named assemblies).

**9. Что такое assembly manifest?**

Любая сборка, статическая или динамическая, содержит коллекцию данных с описанием того, как ее элементы связаны друг с другом. Эти метаданные содержатся в манифесте сборки.

**10.Что такое GAC?**

GAC – Global Assembly Cache

подсистема, хранящая сборки CLI (*CLI assembly*) в централизованном репозитории. Система разработана для борьбы с проблемами разделяемых библиотек, например таких, как [DLL hell](https://ru.wikipedia.org/wiki/DLL_hell).

К сборкам, расположенным в GAC, предъявляется несколько требований. В частности, они должны использовать [strong name](https://ru.wikipedia.org/wiki/Strong_name" \o "Strong name), соблюдать строгую схему указания версий и допускать исполнение нескольких версий кода в рамках единого приложения.

Необходимо, чтобы несколько приложений, использующие одинаковые сборки, могли иметь к ним доступ.

Гарантирует их безопасность от удаления пользователем, так как каталог GAC находится внутри защищенной системной папки Windows.

GAC может хранить несколько разных версий сборок CLI, приложение при запуске запрашивает необходимую версию сборки.

**11.Чем managed code отличается от unmanaged code?**

Managed code – управляемый код, под управлением виртуальной машины.

**12.Как и для чего определен метод Main?**

Метод Main() содержит параметры командной строки, тип возвращаемого значения может быть любой, int или void.

**13.Варианты использования директивы using( using Directive ) в C#.**

Директива using позволяет использовать типы, определенные в пространстве имен, без указания полного пространства имен этого типа. В базовой форме директива using импортирует все типы из одного пространства имен.

Два модификатора : global, static.

**14.Как связаны между собой сборки и пространства имен?**

В программировании на C# пространства имен используются с полной нагрузкой по двум направлениям. Во-первых, платформа .NET Framework использует пространства имен для организации множества классов. Во-вторых, объявление собственного пространства имен поможет в управлении областью действия имен классов и методов в крупных программных проектах.

**15.Что такое примитивные типы данных? Перечислите их.**

**16.Что такое ссылочные типы? Какие типы относятся к ним?**

**17.Какие типы относятся к типам-значениям?**

Структуры, перечисления( числовые типы)

**18. В чем отличие между ссылочными и значимыми типами данных?**

Ссылочные типы бывают только в упакованной форме. Когда переменной значимого типа присваивается другая переменная значимого типа, выполняется копирование всех ее полей. Когда переменной ссылочного типа присваивается переменная ссылочного типа, копируется только ее адрес.

**19.Что такое упаковка и распаковка значимых типов?**

Упаковкой (boxing) называется процесс преобразования типа значения в тип System.Object или в тип интерфейса, который реализуется данным типом значением.

Распаковка - получение указателя на исходный значимый тип (поля данных), содержащийся в объекте

**20.В чем заключается разница между int и System.Int32? double и System.Double и т.д.?**

Криптографический вид. Указание битов.

Разницы никакой, int и Int32 после компиляции в IL код станут одним и тем же. int это ключевое слово для c#,

ровно как Integer для VisualBasic, но оба они представляют собой System.Int32.

int - примитив, доступный в компиляторе C#, в то время как Int32 является типом FCL и следовательно доступен для всех языков,

соответствующих CLS.

В C# int - это simple type, алиаз System.Int32.

C# поддерживает обычный набор типов данных. Для каждого типа данных, который поддерживается C#, существует соответствующий

тип .NET Common Language Runtime. Например, типу int соответствует тип System.Int32. Практически всегда можно использовать

System.Int32 вместо int, но этого не рекомендуется делать, потому что в этом случае код становится более сложным для восприятия.

Просто, double - это алиас класса Double, т.е. в коде пишем double тоже что new System.Double

**21.Для чего используется тип dynamic?**

Отражения или коммуникации с другими компонентами.

может получить какое угодно начальное значение, и на протяжении времени его существования это значение может быть заменено новым

Ключевым моментом использования DLR в C# является применение типов dynamic. Это ключевое слово позволяет опустить

проверку типов во время компиляции. Кроме того, объекты, объявленные как dynamic, могут в течение работы программы менять свой тип.

**22.В чем заключается главное отличие между var и dynamic?**

может получить какое угодно начальное значение, и на протяжении времени его существования это значение может быть заменено новым

В случае ключевого слова var это будет первое присваивание значения переменной, обозначенной ключевым словом var.

После этого присваивания изменить тип значения переменной нельзя. Однако в случае ключевого слова dynamic,

тип переменной можно изменять после первого присваивания.

**23.Что такое неявно типизированная переменная?**

компилятору предоставляется возможность самому определить тип локальной переменной, исходя из значения, которым она инициализируется.

компилятору предоставляется возможность самому определить тип локальной переменной, исходя из значения,

которым она инициализируется. Такая переменная называется неявно типизированной .

**24.Для чего используют Nullable тип?**

это тип, который позволяет представить не только все значения своего нижележащего типа, но и значение null.

**25.Как объявить строковый литерал? Какие операции можно выполнять со строкой?**

**26.Какие есть способы для задания и инициализации строк?**

**27.Какие методы есть у типа String?**

**Compare**

Сравнение двух строк в алфавитном порядке. Разные реализации метода позволяют сравнивать строки и подстроки с учетом и без учета регистра и особенностей национального представления дат и т. д.

**CompareOrdinal**

Сравнение двух строк по кодам символов. Разные реализации метода позволяют сравнивать строки и подстроки

**CompareTo** Сравнение текущего экземпляра строки с другой строкой

**Concat** Конкатенация строк. Метод допускает сцепление произвольного числа строк

**Copy** Создание копии строки 53 **Format** Форматирование в соответствии с заданными спецификаторами формата **IndexOf, LastIndexOf,** … Определение индексов первого и последнего вхождения заданной подстроки или любого символа из заданного набора Insert Вставка подстроки в заданную позицию **Join** Слияние массива строк в единую строку. Между элементами массива вставляются разделители (см. далее) **Length** Длина строки (количество символов) Remove Удаление подстроки из заданной позиции **Replace** Замена всех вхождений заданной подстроки или символа новой подстрокой или символом **Split** Разделение строки на элементы, используя заданные разделители. Результаты помещаются в массив строк **Substring** Выделение подстроки, начиная с заданной позиции

**28.В чем отличие пустой и null строки?**

null - отдельный тип

пустая строка - string

**29.Как можно выполнить сравнение строк?**

String.Compare Сравнивает значения двух строк. Возвращает целочисленное значение.

String.CompareOrdinal Сравнивает две строки без учета локального языка и региональных параметров. Возвращает целочисленное значение.

String.CompareTo Сравнивает текущий строковый объект с другой строкой. Возвращает целочисленное значение.

String.StartsWith Определяет, начинается ли строка с переданной строки. Возвращает логическое значение.

String.EndsWith Определяет, заканчивается ли строка переданной строкой. Возвращает логическое значение.

String.Contains Определяет, встречается ли символ или строка в другой строке. Возвращает логическое значение.

String.Equals Определяет, совпадают ли две строки. Возвращает логическое значение.

**30.В чем отличие типов String и StringBuilder?**

В отличие от типа string, в классе StringBuilder можно изменять отдельный символ по его индексу. Например.

// Изменить отдельный символ строки можно в отличие

от типа String StringBuilder sb = new StringBuilder("Text."); sb[0] = 'N'; Console.

**31.Поясните явные преобразования переменных с помощью команд Convert.**

**32.Как выполнить консольный ввод/вывод?**

Console.WriteLine. - вывод

Console.ReadLine(). - ввод

**33.Приведите примеры определения и инициализации одномерных и двумерных массивов. 34.Что такое ступенчатый массив? Как его задать?**

Ступенчатый массив представляет собой массив массивов, в котором длина каждого массива может быть разной.

Следовательно, ступенчатый массив может быть использован для составления таблицы из строк разной длины.

Ступенчатые массивы объявляются с помощью ряда квадратных скобок, в которых указывается их размерность.

**35.Какие типы можно использовать в foreach? Приведите пример.**

Можно использовать типы, которые реализуют интерфейс System. Collections. IEnumerable или System.

**36.Что такое кортеж? Для чего и как он используется?**

Кортежи (tuple) комбинируют объекты различных типов (от одного до восьми). Типы и выражения.

для группирования нескольких элементов данных в упрощенную структуру данных

Кортеж представляет набор значений, заключенных в круглые скобки

**37.Что такое локальная функция? Какова область ее видимости?**

Локальные функции представляют функции, определенные внутри других методов. Локальная функция, как правило, содержит действия, которые применяются только в рамках ее метода.

Локальные функции представляют функции, определенные внутри других методов. Локальная функция, как правило,

содержит действия, которые применяются только в рамках ее метода.

**38. В чем разница между кодом, заключенным в блок checked и кодом, заключенным в блок unchecked?**

Так, если требуется указать, что выражение будет проверяться на переполнение, следует использовать ключевое слово **checked**, а если требуется проигнорировать переполнение — ключевое слово **unchecked**.

**39.Какой контекст (checked/unchecked) применяется по умолчанию? Как можно переопределить это поведение?**

Unchecked по умолчанию. Его можно изменить в настройках Vs

**40.Для чего используется ключевое слово fixed? Каковы особенности его использования.**

Ключевое fixed слово позволяет закрепить локальный элемент в стеке, чтобы предотвратить его сбор или перемещение во время сборки мусора.

Адрес фиксированной или закрепленной переменной не изменяется в течение длительности инструкции. Объявленный указатель можно использовать только в соответствующей fixed инструкции. Объявленный указатель доступен только для чтения и не может быть изменен.

Ключевое fixed слово позволяет закрепить локальный элемент в стеке, чтобы предотвратить его сбор или

перемещение во время сборки мусора. Он используется для низкоуровневых сценариев программирования.