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

.NET Framework — программная платформа. Основой платформы является общеязыковая среда исполнения Common Language Runtime (CLR), которая подходит для разных языков программирования. Функциональные возможности CLR доступны в любых языках программирования, использующих эту среду.

*2. Что такое CLR, FCL/BCL, CLI, IL?*

1. Общеязыковая среда CLR решает задачи автоматического выявления типов .NET, загрузки этих типов и управление ними. Среда CLR осуществляет управление памятью, обслуживание приложения, обработку потоков и реализует многочисленные проверки связанные с безопасностью.

2. CLI – интерфейс командной строки.

3. (Intermediate Language – IL) – промежуточный язык, на котором написаны файлы с расширением .dll .exe

4. BCL — стандартная библиотека классов платформы «.NET Framework»

*3. Пояснить работу JIT-компилятора?*

JIT-компиляция-- технология увеличения производительности программных систем, использующих байт-код, путём компиляции байт-кода в машинный код или в другой формат непосредственно во время работы программы. Таким образом достигается высокая скорость выполнения.

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

Common Type System (сокр. CTS, рус. Общая система типов) — часть .NET Framework, формальная спецификация, определяющая, как какой-либо тип (класс, интерфейс, структура, встроенный тип данных) должен быть определён для его правильного выполнения средой .NET

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

Корень всех типов: System.Object. позволяет гарантировать наличие у каждого типа минимального набора функциональных возможностей. Все типы получают «бесплатно» четыре открытых метода: bool, int, type,string.

*6. Что находится в mscorlib dll?*

Большинство классов, интерфейсов, перечислений, структур, делегатов.

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

Сборка (assembly) — 1) это абстрактное понятие, для логической группировки одного или нескольких управляемых модулей или файлов ресурсов.

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.

*8. Какие виды сборок существуют?*

с нестрогими именами (weakly named assemblies)

со строгими именами (strongly named assemblies).

*9. Что такое assembly manifest?*

описывает элементы сборки взаимодействующие друг с другом

*10.* *Что такое GAC?*

(Глобальный кеш сборок)- подсистема, хранящая сборки CLI (CLI assembly) в централизованном репозитории.

*11. Чем managed code отличается от unmanaged code*

Управляемый код — термин, введённый фирмой Microsoft, для обозначения кода программы, исполняемой под «управлением» виртуальной машины .NET— Common Language Runtime или Mono. При этом машинный код называется неуправляемым кодом.

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

Main() — это главный метод программы, ее входная точка, является частью класса. Класс, в котором находится этот метод, по умолчанию называется Program. Это статический метод, имеет ключевое слово static, поэтому он реализуется еще до создания объекта класса, в котором он находится.

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

Директива using разрешает использование типов в пространстве имен, поэтому уточнение использования типа в этом пространстве имен не требуется.

Также директива using позволяет создавать псевдонимы пространства имен или типа.

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

Пространство имен и сборка (файл, в котором реализован тип) не обязательно связаны друг с другом. В частности, различные типы, принадлежащие одному пространству имен, могут быть реализованы в разных сборках.

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

Примитивный тип — тип данных, предоставляемый языком программирования как базовая встроенная единица языка.

bool,byte,sbyte,short,ushort,int,uint,long,ulong,float,double,decimal,string,char,object.

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

Ссылочные типы – хранят в себе ссылку на данные.

• Ссылочные типы: Тип object

• Тип string

• Классы (class)

• Интерфейсы (interface)

• Делегаты (delegate)

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

• Целочисленные;

• С плавающей запятой;

• Десятичный;

• Логический;

• Структуры;

• Перечисления.

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

Ссылочные хранят в себе ссылку на данные, а значимые – сами данные.

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

Это перенос их значений из стека в кучу.

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

Разницы нет нигде.

*21.Для чего используется тип dynamic?*

Это ключевое слово позволяет опустить проверку типов во время компиляции.

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

Var позволяет создать переменную с неявным типом данных, но она должна быть сразу инициализирована и тогда получит самый подходящий тип данных в зависимости от своего значения. У переменной, созданной с помощью dynamic может изменятся тип данных в течение выполнения программы.

*23.Что такое неявно типизированная переменная?*

Переменная, которой на задан тип данных, создаётся при помощи var.

*24.Для чего используют Nullable тип?*

Для того, чтоб показать, что переменная может принимать значение null.

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

Строковы литерал можно объявить либо при помощи Console.Writeline(), либо string. Строки можно сцепливать, копировать, выделять подстроки, разделять строки на слова, вставлять строку в другую на выбранную позицию, удалять элементы строки.

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

Создавать строки можно, как используя переменную типа string и присваивая ей значение, так и применяя один из конструкторов класса String

*27.Какие методы есть у типа String?*

Join() – объединение строк, Compare() сравнение строк, IndexO() – поиск первого вхождения выбранного символа, Split() – разделение, Trim() – обрезка,Insert() – вставка, Remove() – удаление,Replace() – замена, ToLower,ToUpper – смена регистра.

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

С пустой строкой могут проводится действия, как с обычной, у неё есть своя длина. Null строка ничего не имеет, и операций с ней провести нельзя.

*29.Как можно выполнить сравнение строк?*

При помощи Compare().

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

StringBuilder позволяет создать строку с динамической длинной, String такого не может позволить. В String есть методы, которые отсутствуют в StringBuilder.

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

Convert позволяет преобразовывать один тип данных в другой, но есть и исключения. При успешном явном преобразовании, если необходимо, Convert удаляет те данные, что не помещаются в новый тип.

*32.Как выполнить консольный ввод/вывод?*

Ввод – Console.ReadLine()

Вывод – Console.WriteLine()

Ввод кнопки – Console.ReadKey()

*33.Приведите примеры определения и инициализации одномерных и двумерных массивов.*

Одномерный: int[] mass1 = {1,2,3,4,5};

Двумерный: int[,] mass2 ={{2,3},{4,5}};

*34.Что такое ступенчатый массив? Как его задать?*

Ступенчатый массив – двумерный массив, если который представить в виде таблицы, то его столбцы будут иметь неодинаковую длину.

int[][] mass3 = new int [3][];

mass3[0] = new int[3];

mass3[1] = new int[4];

mass3[2] = new int[2];

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

Foreach можно применять к коллекциям, например: массивы, очереди, хеш-таблицы и т.д.

Foreach(int i in array1){

Console.WriteLine(i)}// выведет все элементы в этом массиве

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

Кортежи предоставляют удобный способ для работы с набором значений, который был добавлен в версии C# 7.0.Кортеж представляет набор значений, заключенных в круглые скобки. Кортеж можно использовать для получения из функции несколько переменных разных типов данных. Также их можно передавать в функцию.

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

Локальные функции представляют функции, определенные внутри других методов.

Может вызываться только в той ф-ции, в которой объявлена.

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

Код в блоке checked будет проверяться на переполнение а переменных, а с unchecked – наоборот.

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

По умолчанию применяется контекст checked…

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

Fixed указывает сборщику мусора не перемещать объект. Это означает, что позиция переменной в памяти в памяти фиксируется, чтобы на нее мог ссылаться указатель.