Тест 1

1. Что из перечисленного не является парадигмой программирования?

Нормативное программирование

2. Что такое инкапсуляция?

размещение в одном компоненте данных и методов, которые с ними работают.

3. Что такое класс?

элемент, описывающий абстрактный тип данных и его частичную или полную реализацию.

4. Какой классификации типов данных НЕ существует?

По назначению.

5. Зачем нужен модификатор доступа?

для задания области видимости членов классов.

6. что такое парадигма программирования

Совокупность идей

7. что такое абстракция?

использование только тех

8. Что такое объект?

Некоторая сущность в цифровом пространстве

9. Зачем нужен тип данных?

Все перечисленное

10. на что указывает применение модификатора доступа public к элементу класса?

Элемент класса доступен всюду

11. в чем заключается структурное программирование?

Парадигма программирования, в основе которой лежит представление программы в виде иерархической структуры блоков

12. какого элемента НЕ может быть в классе?

Объект

13. что такое упаковка?

Процесс преобразования типа значения в тип object

14. зачем нужен модификатор доступа?

Для задания области видимости членов классов

**Тест 2**

1. **что такое деструктор?**

особый метод класса, используемый для очистки полей класса

1. **какой вид наследования относится к типу «реализация»?**

от интерфейса

1. **как скрыть метод базового класса в дочернем классе?**

в дочернем классе у метода добавить модификатор new

1. **как происходит работа с абстрактными методами в классе-наследнике?**

класс обязан определить все абстрактные методы, используя ключевое слово override

1. **какая особенность отличает наследование от интерфейса от иных способов наследования?**

может наследоваться сразу же от нескольких интерфейсов

1. **что понимается под термином «наследование» в программировании?**

процесс, посредством которого один объект может приобретать свойства другого

1. **какие есть особенности у абстрактного класса по сравнению с обычным?**

от абстрактного класса нельзя создать объект

1. **что такое конструктор?**

особый метод класса, используемый для инициализации полей классе

1. **чем абстрактный класс отличается от обычного класса?**

в абстрактном классе могут быть методы, у которых нет реализации

1. **что такое интерфейс?**

ссылочный тип, который состоит только из абстракций

1. **на что указывает применение модификатора доступа protected к элементу класса?**

элемент класса доступен только в классе и его производных

1. **какое из этих утверждений ложно?**

класс можно наследовать от нескольких абстрактных классов и интерфейсов

1. **чем абстрактный класс отличается от обычного класса?**

в абстрактном классе могут быть методы, у которых нет реализации

1. **как правильно переопределить метод в дочернем классе?**

в родительском классе у метода добавить модификатор virtual, а в дочернем – override

1. **Какие особенности у конструктора**

Создаётся в скрытом виде, если явно не указан, отсутствует тип возвращаемого значения, название идентично названию класса, в котором он задаётся

1. **На что указывает применение модификатора доступа protected к элементу класса?**

Элемент класса доступен только в классе и его производных

1. **Через какое ключевое слово можно обращаться к элементам родительского класса в дочернем классе?** BASE

**Тест 3**

1. **Каким способом нельзя перегрузить метод?+**

Разным типом возвращаемого значения

1. **В чем заключается параметрический полиморфизм?+**

Позволяет определить метод…..

1. **Что такое «метод расширения»?+**

Подход позволяющий добавлять новые методы в уже существующие…

1. **Что такое полиморфизм?+**

Возможность объектов с одинаковой спецификацией…

1. **Какие виды приведения типов существуют?+**

Явные и неявные

1. **Что означает применение ограничения «struct» на параметр?+**

Аргумент типа должен быть типом значения

1. **Что такое «позднее связывание»?+**

Связь между местом вызова метода и непосредственно местом, где этот метод реализован на этапе выполнения программы

1. **Какого вида полиморфизма не существует?+**

Расширенный полиморфизм

1. **Что из ниже перечисленного НЕ относится к специализированному полиморфизму?+**

Перегрузка типов

1. **Какие виды параметрического полиморфизма не существует?+**

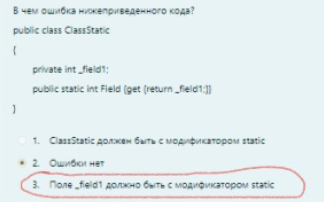
Параметрические поля

1. **Что такое "ранее связывание"?+**

связь между местом вызова метода и непосредственно местом, где этот метод реализован на этапе компиляции

1. **В чём ошибка нижеприведённого кода? Public class ClassStatic{ public int \_field1; public static int Field{get {return \_field1;}}**

Поле \_field1 должно быть с модификатором static

1. +

**Тест 4**

1. **Что такое массив массивов?**

Массив массивов представляет собой одномерный массив, в котором каждый элемент является массивом

1. **По способу доступа к элементу коллекции делются на:**

Индексируемые и ассоциативные

1. **К какому типу коллекций отнести коллекцию List <T>?**

Индексируемая, универсальная

1. **Что такое Hashtable?**

Неуниверсальный класс, который представляет коллекцию пар «ключ-значение», которые упорядочены по хэш-коду ключа

1. **Что такое индексатор?**

Особые элементы класса, позволяющие обращаться к экземпляром класса или структуры точно так же, как и к массивам

1. **Какое из этих утверждений ложно?**

Массив может в себе одновременно хранить элементы как ссылочных типов, так и типов значений

1. **Какого типа коллекции не существует?**

расширенные

1. **Что такое множество?**

Коллекция, которая не содержит повторяющихся элементов

1. **Что такое Dictionary <TKey, TValue>?**

Представляет собой набор пар ключ-значение. Каждый ключ коллекции должен быть уникальным

1. **Какая их этих структур не является предшественником кортежей?**

(int, string, int)

1. **Что такое кортеж?**

Краткий синтаксис для группирования нескольких элементов данных в упрощенную структуру данных

1. **Что такое Хеш-функция?**

Функция, осуществляющая преобразование массива входных данных произвольной длины в выходную битовую строку установленной длины, выполняемое определенным алгоритмом

13. **Что такое очередь?**

Представляет коллекцию объектов, основанную на принципе “первым вошел – первым вышел”

14. **По типу хранимых данных коллекции делятся на:**

Универсальные и неуверсальные

15. **Какое из этих утверждений истино?**

В памяти массив располагается единым целым

**ТЕСТ 5**

**1. Что такое ковариантность?**

К делегату можно привязать метод возвращаемый тип которого является производным ( класс – наследник ) от типа…

**2. Какое из нижеперечисленных описаний относится к встроенному делегату Action?**

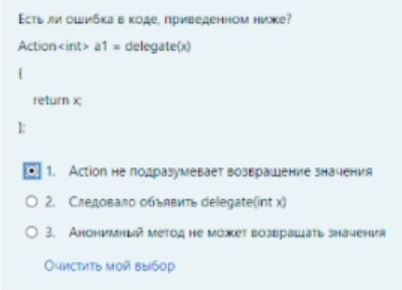
Принимает до 16 параметров и не возвращает значение

**3. Какое из нижеперечисленных описаний относится к событийному программированию?**

Парадигма программирования, в которой выполнение программы определяется действиями пользователя…

1. **Есть ли ошибка в коде, приведенном ниже? Action <int> a1 = delegate(x){ return x; };**

Action не подразумевает возвращение значения



**5. Какое правило для передаваемых параметров ошибочно?**

Скобки должны быть всегда

**6. Какое из нижеперечисленных описаний относится к встроенному делегату Predicate?**

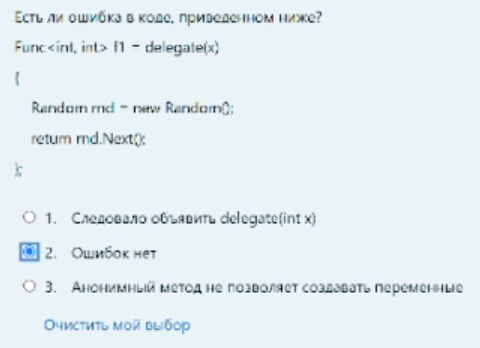
Используется для сравнения объекта T по определенному условию

**7. Что такое событие?**

Сообщение, которое возникает в различных точках исполняемого кода при выполнении…

1. **Есть ли ошибка в коде, приведённом ниже? Func <int, int> f1 = delegate(x) { Random rnd = new Random(); return md.Next(); }};**

Ошибок нет



**9. Какое правило для тела оператора ошибочно?**

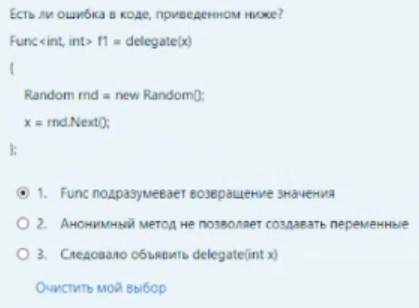
Тело лямбды оператора должно состоять не более чем из 3-х операторов

**10. Кто такой издатель?**

Класс, отправляющий событие

1. **Есть ли ошибка в коде, приведённом ниже? Func <int, int> f1 = delegate(x) { Random rnd = new Random(); x=rnd.Next();};**

Func подразумевает возвращение значения



**12. Что такое лямбда-выражение?**

Анонимная функция, с помощью которой можно создавать упрощенную запись…

**13. Что такое многоадресность?**

При привязке метода к объекту-делегату в объекте сохраняется адрес привязываемого метода. К объекту-делегату можно привязать сразу несколько методов. Это называется многоадресностью.

Чтобы добавить в список методов делегата (список вызова) дополнительный метод, необходимо просто добавить его к делегату с помощью оператора сложения или назначения сложения ("+" или "+=").

**14. Что такое делегат?**

Тип, который представляет собой ссылки на методы с определенным списком параметров и возвращаемым типом

**15. Какое из перечисленных описаний относится к встроенному делегату Func?**

Принимает до 16 параметров и возвращает значение

**ТЕСТ 6**

**1. Что такое поток?**

Абстракция последовательности байтов

**2. За что отвечает параметр buffer в методах чтения и записи потоков?**

Массив байтов, из которого (в который) байты копируются в (из) в поток

**3. Какой кодировки не существует**

Code-32

**4. Какой класс отвечает за работу с сетевым потоком?**

NetworkStream

**5. Какое описание относится к классу Directory?**

Предоставляет статические методы для создания, перемещения и перечисления в каталогах и вложенных каталогах удаления, перемещения и открытия одного файла.

**6. Метод чтения int Read (….). За что отвечает параметр offset?**

Смещение байтов (начиная с нуля) в buffer, с которого начинается сохранение данных, считанных из текущего потока

**7. Как должен открываться файл при выборе значения перечисления FileMode OpenOrCreate?**

Открыть файл, если он существует, в противном случае должен быть создан новый файл

**8. Какой класс отвечает за возможность отправки зашифрованного потока байт?**

CryptoStream

**9. Какое описание относится к классу Fileinfo?**

Предоставляет свойства и методы экземпляра для создания, копирования, удаления, перемещения и открытия файлов.

**10. Какой основной класс для работы с потоками в C#?**

Stream

**11. За что отвечает перечисление FileMode?**

Указывает, каким образом операционная система должна открыть файл

**12. Какое описание относится к классу File?**

Предоставляет статистические методы для создания, копирования, удаления, перемещения и открытия одного файла

**13. Метод записи void Write … Что означает параметр count?**

Максимальное количество байтов, которое должно быть записано в текущий поток

**14. Какой класс отвечает за работу с поток памяти?**

MemoryStream

**15. Какая операция Не включается в поток?**

Фильтрация

**ТЕСТ 7**

**1. Что такое штатные ошибки?**

Это ошибки, обработка которых является частью бизнес логики

**2. За что отвечает блок try?**

Блок, в котором располагается код, выполнение которого считается потенциально опасным

**3. Какое из ниже перечисленных утверждений относится к термину «иерархия исключений» в языке С#?**

Базовым для всех типов исключений является тип Exception

**4. Какое описание соответствует классу ошибок NullReferenceExeption?**

Исключение выбрасывается при вызове функции или доступа к полям нулевого объекта.

**5. К какому уровню логирования относится описание: обычные сообщения, информирующие о действиях системы?**

Info

**6. За что отвечает блок finally?**

Блок, в котором располагается код, который должен…

**7.** **Какое из ниже перечисленных утверждений относится к термину «иерархия исключений» в языке Java?**

У всех исключений есть общий класс-родитель Throwable

**8. К какому уровню логирования относится описание: ошибка в работе системы, требующая вмешательства?**

Error

**9.** **Какое из ниже перечисленных утверждений относится к термину «иерархия исключений» вне зависимости от языка?**

Определяет порядок следования исключения в блоках catch

**10. Какое описание соответствует классу ошибок IndexOutOfRangeException?**

Исключение выбрасывается, когда происходит попытка получения…

**11. За что отвечает блок catch?**

Блок, в котором располагается код обработки возможных ошибок

**12. Что такое непредвиденные ошибки?**

Это ошибки, появление которых не ожидается

**13. К какому уровню логирования относится описание: сообщение отладки, профилирования?**

Debug

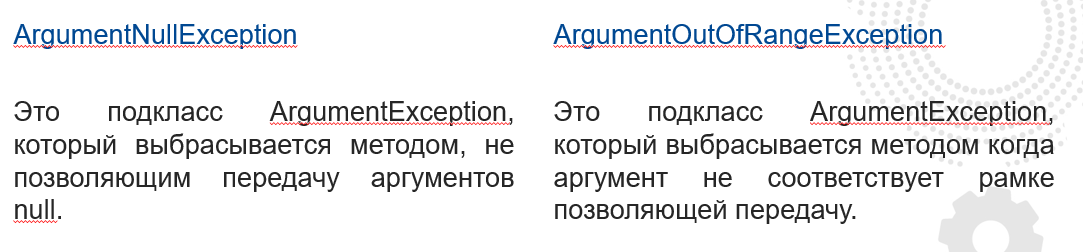
**14. Какое описание соответствует классу ошибок InvalidOperationException?**

Выбрасывается методом при недействительном статусе

**15. Что такое исключение?**

Механизм, который позволяет программе обрабатывать нетипичную ситуацию и при этом не прекращать работу

ArgumentException - Основной класс всех исключений связанных с аргументом (Argument).



Warn: записывая такое сообщение, система пытается привлечь внимание. Произошло что-то странное. Следует разобраться в том, что произошло, что это означает, и отнести ситуацию либо к инфо-сообщению, либо к ошибке. Соответственно, придётся доработать код обработки таких ситуаций.

Fatal: это особый класс ошибок. Такие ошибки приводят к неработоспособности системы в целом, или неработоспособности одной из подсистем. Чаще всего случаются фатальные ошибки из-за неверной конфигурации или отказов оборудования. Требуют срочной, немедленной реакции.

**ТЕСТ 8**

**1. Когда выбирать интерфейс IComparable?**

Если структура данных имеет естественную операцию сравнения

**2. Что такое управляемый ресурс?**

Ресурсы, которые представляют собой чистый код .NET и управляются средой выполнения и находятся под ее прямым контролем

**3. Какого метода не объявлен в интрефейсе IEnumerator?**

GetEnumerator

**4. Что такое «поверхностное копирование»?**

Копирование, при котором НЕ создается копий ссылочных полей клонируемого объекта

**5. Когда выбирать интерфейс IEqutable?**

Если есть только один вариант проверки двух экземпляров T на равенство…

**6. Какое из ниже приведенных утверждений ложно?**

Большие объекты помещаются в кучу больших объектов, которые называют поколение 0.

**7. Зачем нужен интерфейс IEnumerator?**

Поддерживает простой перебор элементов универсальной коллекции

**8. Что такое глубокое копирование?**

Копирование, при котором создаются копии ссылочных полей клонируемого объекта.

**9. Когда выбирать интерфейс IComparer?**

Если требуется реализовать собственный сортировки, для конкретного случая, учитывая конкрутную ситуацию, а не суть классов

**10. Когда выбирать интерфейс IEqualityComparer?**

Если есть несколько одинаковых разумных методов сравнения двух T на равенство

**11. Что такое неуправляемые ресурсы?**

те ресурсы, которые относится к пустыне за пределами изолированной программной среды .NET.

**12. Для IEquatable не требуется реализовывать метод Equals(T t1, T t2)**

**13. Для IComparable не требуется реализовывать Compare( T t1, T t2)**

**14.** **Для IEnumarator не требуется реализовывать GetEnumerator()**

**15. Как работает метод MemberwiseClone?**

Создает неглубокую копию, создавая новый объект, а затем копируя нестатические поля текущего объекта в новый объект.

**16. Зачем нужен интерфейс IEnumerable?**

Предоставляет перечислитель, который поддерживает простой перебор элементов в указанной коллекции.