

Вопросы по дисциплинам

“Объектно-ориентированное программирование”

1-40 05 01 03 «Информационные системы и технологии (издательско-полиграфический комплекс)»

1-47 01 02 «Дизайн электронных и веб-изданий»

” Современные технологии программирования мобильных систем”

1-98 01 03 Программное обеспечение информационной безопасности мобильных систем

“Объектно-ориентированные технологии программирования и стандарты проектирования”

1- 40 01 01- Программное обеспечение информационных технологий (специализация
«Программирование интернет-приложений»)

2018/2019 гг.

1. Системы контроля версий. Классификация. Git, основные возможности
2. Состав. NET Framework. Структура среды выполнения CLR.
3. Структура управляемого модуля - portable executable (PE). Понятие и исполнение сборки. CIL.
4. CTS (Common Type System). Типы данных C#. Ссылочные и типы значений.
5. Понятие упаковки и распаковки типов. Типы Nullable: преобразование, проверка, null-объединение
6. Тип данных String: операции, литералы, пустые и нулевые строки, форматированный вывод.
7. Неявная типизация – назначение и использование.
8. Массивы C# одномерные, прямоугольные и ступенчатые.
9. Понятие кортежей. Свойства, создание
10. Принципы объектно-ориентированного программирования.
11. Класс. Элементы класса. Свойства и индексаторы.
12. Класс. Константы. Поля только для чтения. Инициализаторы класса.
13. Спецификаторы доступа C#. Видимость типов. Доступ к членам типов.
14. Класс. Конструкторы и их свойства. Деструкторы
15. Класс и методы System.Object.
16. Статические методы и статические конструкторы класса.
17. Статические классы. Методы расширения и правила их определения.
18. Анонимные типы.
19. Модификаторы параметров - ref , out, params. Необязательные и именованные аргументы.
20. Перегрузка методов и операторов. Правила перегрузки операторов.
21. Операции преобразования типа. Явная и неявная форма. Ограничения.
22. Вложенные типы. Вложенные объекты
23. Правила наследования C#.
24. Соккрытие имен при наследовании. Обращение к скрытым членам
25. Использование операций is и as
26. Полиморфизм. Виртуальные методы, свойства и индексаторы. Правила переопределения.
27. Понятие раннего и позднего связывания.
28. Абстрактные классы и методы. Бесплодные классы.
29. Структур в C#.
30. Интерфейсы. Свойства интерфейсов. Реализация интерфейсов.
31. Явная и неявная реализация интерфейсов. Работа с объектами через интерфейсы.
32. Ковариантность интерфейсов. Контравариантность интерфейсов
33. Стандартные интерфейсы .NET. Назначение и применение.
34. Исключительные ситуации. Генерация и повторная генерация исключений.
35. Исключительные ситуации. Варианты обработки исключений. Фильтры исключений
36. Обобщения (generics). Свойства обобщений.
37. Концепция ограничений обобщений. Статические члены обобщений.
38. Делегаты. Определение, назначение и варианты использования. Обобщенные делегаты.

39. Анонимные функции. Лямбда-выражения.
40. Обобщённые делегаты .NET. Action, Func, Predicate
41. События и делегаты.
42. Стандартные коллекции .NET. Типы коллекций.
43. Стандартные интерфейсы коллекций.
44. IEnumerable и IEnumerator
45. LINQ to Objects. Синтаксис. Форма. Возврат результата. Грамматика выражений запросов. Отложенные и неотложенные операции.
46. LINQ to Objects. Операции Where, Select, Take, OrderB, Join, GroupBy
47. Рефлексия. System Type.
48. Классы для работы с файловой системой.
49. Синтаксическая конструкция using. Чтение и запись файлов. Потокоточные классы.
50. Классы адаптеры потоков.
51. Сериализация. Форматы сериализации.
52. Сериализация контрактов данных. интерфейс ISerializable.
53. Атрибуты. Создание собственного атрибута.
54. Процесс. Домен приложений. Поток выполнения.
55. Создание потоков , классы приоритетов. Состояния потоков
56. Синхронизация потоков. Lock. Monitor. Mutex. Semaphore
57. Библиотека параллельных задач TPL. Класс Task. Состояние задачи.
58. Способы создания Task. Возврат результата. Отмена выполнения задач. Продолжения.
59. Параллелизм при императивной обработке данных. Класс Parallel/
60. Асинхронные методы. async и await

Темы задач

С# базовый	<p>Типы, кортежи, класс, структуры, конструкторы свойства, индексаторы, константы, доступность, переопределения, преобразования, массивы, наследование, интерфейс, станд. интерфейсы, статика, виртуальность, коллекции, делегаты, стандартные делегаты, события, обобщения, исключения, методы расширения, перегрузка операций, <i>анонимные функции, лямбда выражения, LINQ, String, потоковые классы, классы для работы с файловой системой, Сериализация, потоки – управление и синхронизация, Task</i> .</p> <p><i>Пример задачи:</i></p> <p style="text-align: center;">№ 1</p> <p>Создать класс Circle с закрытым полем - радиус и методом вычисления площади круга. Переопределить ToString и Equals. Перегрузить метод CompareTo стандартного унаследованного интерфейса IComparable, который сравнивает круги по площади. Создать List кругов и отсортировать их и вывести на консоль.</p> <p>Создать класс PDate с полями: день (1-31), месяц (1-12). Класс имеет два конструктора и метод NextDay(), который увеличивает день (корректно). Свойства полей класса должны проверять корректность задаваемых параметров. Перегрузить оператор >, сравнивающий две даты. Используя наследование от PDate создайте класс PMDate в котором добавляется год. Переопределите NextDay(). Написать демонстрацию создания объектов всех классов с вызовом методов, свойств и оператора.</p>
------------	--