

БИЛЕТЫ ОАиП - С#

1. Переменные, ввод\вывод, математика, условия

- Что такое переменная и как придать ей какое-то значение?
- Назовите методы для вывода и ввода в консоль.
- Конвертация. Как происходит конвертация текста в целочисленное число?
- Условные конструкции. Как задать условие? Как можно заменить множество условий, если там везде проверяется значение одной переменной?

2. Циклы while, for, do while

- Циклы. Виды циклов. Для чего используют циклы?
- Цикл for. К какому виду циклов он относится, принцип работы, структура объявления цикла.
- Цикл while. К какому виду циклов он относится, принцип работы, в чем его отличие от for?
- Цикл do...while. К какому виду циклов он относится, чем отличается от while и for?

3. Коллекции: массив, лист, матрица. Перебор с помощью foreach

- Коллекция. Виды коллекций.
- Как объявляется массив? Индексы и получение элементов массива.
- Структура объявления листа. В чем отличие и преимущество листа от массива?
- Структура матрицы. В чем преимущество матрицы и как обратиться к элементу матрицы?
- Цикл foreach. Принцип работы и его структура. Чем отличается foreach от for?

4. Методы: обычные, возвращаемые, с параметрами

- Метод. Для чего нужны методы и как к ним обращаться?
- Параметры метода. Как передать значения из одного метода в другой? Можно ли дать параметрам значения по умолчанию?
- Возвращаемые методы. Что такое «void»? Что нужно сделать для того, чтобы метод не просто хранил в себе код, а возвращал значение из этого кода?

5. Linq запросы с коллекциями, тернарные выражения, проверки на null(пустоту)

- Как можно быстро найти минимальный элемент в коллекции? Какие еще методы позволяют быстро работать с коллекциями? Как эти методы называются?
- Сложные методы. можно ли вставить в эти методы условия? как это записать? как называется выражение внутри такого метода?
- Какое значение по умолчанию принимает переменная? Структура проверки на пустоту.
- Что такое тернарное выражение? Для чего используют тернарные выражения? Сколько их всего? Какие символы используются, чтобы сделать тернарное выражение?

6. Чтение символов с клавиатуры

- Как ввести какую-то системную клавишу в консоль и прочитать ее?
- Условия для клавиши. Расскажите структуру создания условий для клавиши.

- Сочетания клавиш. Как сделать так, чтобы можно было создать какую-то комбинацию клавиш?
- 7. Расположение курсора в консоли, очистка консоли, стрелки**
- Какой метод используется для того, чтобы очистить ненужную или повторяющуюся информацию?
 - Расскажите о методе, который может изменить расположение нашего текста в консоли. Как сделать так, чтобы условно расположение нашего текста изменялось по нажатию на клавишу?
- 8. Создание своих типов данных, var, преобразование данных**
- Для чего нужно создавать свой тип данных? Как создать свой тип данных?
 - Использование нового типа данных. Как создается сложная переменная? Как придать нашим новым переменным какое-то значение и вывести их в консоль?
 - Для чего нужна коллекция с типом данных и как ее создать? Что нам нужно сделать чтобы вывести содержимое коллекции в консоль?
 - Что такое var? Можно ли объявить переменную с помощью var, но не придать ей значение?
 - Может ли код конвертировать один тип данных в другой сам по себе? Что такое неявное и явное преобразование?
 - С помощью скольких способов мы можем выполнить приведение? Расскажите про структуру каждого.
- 9. Классы как контейнеры, модификатор доступа, инкапсуляция, статичные объекты**
- Что можно хранить в классах? как вы понимаете классы, если использовать их не как свой тип данных?
 - Как взаимодействовать с элементами одного класса в другом классе? видимы ли эти элементы в другом классе по умолчанию?
 - Что такое конструктор? Зачем он нужен? Можно ли делать несколько конструкторов одновременно?
 - Что такое глобальная переменная и как ее создать?
 - Что такое static и зачем он нужен?
 - Структура вызова статичного объекта. В каких случаях класс не может быть статичным?
 - Примеры статичных и обычных классов.
- 10. Работа с файлами: сохранение в текст**
- Чтение файла. Как прочитать содержимое в файле и вывести это в консоль? Что делать, если файл, который мы хотим прочитать, не будет существовать?
 - Запись в файл. Как записать некое содержимое в файл? Что будет, если файл, в который мы хотим записать информацию, не будет существовать?
 - Как записать новое содержимое в файл, не удаляя прежнее содержимое?
- 11. Работа с файлами: JSON**
- Структура JSON. Что поменяется в структуре JSON, если в листе будет несколько переменных?

- Что такое сериализация? Как сериализовать лист с переменными в файл?
- Что такое десериализация? Расскажите про структуру десериализации. Как вывести в консоль содержимое, которое мы десериализовали?

12. Работы с файлами: XML

- Структура XML. Что поменяется в структуре XML, если в листе будет несколько переменных?
- Сериализация. С помощью какого типа данных мы будем сериализовывать наш класс? Как указать, что сериализовывать мы будем именно класс? Расскажите структуру сериализации в файл XML. Что такое файловый поток?
- Десериализация. Структура десериализации. В чем отличие структуры десериализации от структуры сериализации?

13. Работа с папками: создание, чтение данных, запуск процессов

- Структура создания папки. Как создать в одной папке еще несколько папок?
- Структура удаления папки. Что делать, если мы хотим удалить папку, которой не существует? Что нужно сделать для того, чтобы удалить папку, в которой есть файлы?
- Запуск файлов. Структура запуска файла. Можно ли запустить файл используя только путь до него?

14. Try-catch-finally, обработка ошибок

- Для чего используют конструкцию try-catch-finally? Расскажите о принципе работы try-catch.
- С помощью чего можно вывести ошибку в консоль? Как обработать какую-то определенную ошибку?
- Для чего нужен блок finally?

15. Многопоточность, Thread

- Что такое многопоточность? С помощью какой команды останавливается поток?
- Как создать поток? Какие способы создания потока вы знаете? Расскажите про структуру каждого.
- Как называется команда, которая запускает поток? Какие дополнительные методы потоков вы знаете помимо остановки/запуска?

16. Тесты, NUnit, Assert

- Как создать и подключить тестовый проект к основному приложению? Как запускать тесты?
- Какие атрибуты должны стоять над методами, чтобы программа поняла, что это тест?
- При помощи какого класса происходит тестирование? Перечислите минимум 4 метода, которыми можно воспользоваться через этот класс.

17. Перечисляемые типы, enum

- Что такое enum, для чего он нужен и какая у него структура?
- Начиная с какого числа у нас идет нумерация наших объектов в enum? Можем ли мы задавать собственные порядковые значения для переменных внутри enum?

- Как вывести какой-то определенный объект в консоль, с какой ошибкой мы можем столкнуться и как ее решить?

18. Наследование, базовые классы

- В каких случаях можно использовать наследование? Что такое родительский и дочерний класс? Как наследовать класс?
- Наследуются ли все объекты из родительского класса в дочерний? Нужно ли их пересоздавать в дочернем классе?
- Что такое upcast и downcast? В чем их разница? Могу ли я сделать upcast без приведения данных?

19. Интерфейсы

- Что такое интерфейс? Как правильно назвать интерфейс в коде?
- Обязательна ли реализация каждого метода интерфейса, если класс наследует этот интерфейс?
- Если класс наследует интерфейс, можно ли реализовать upcast и downcast к этому интерфейсу?

20. Полиморфизм

- Для чего нужен полиморфизм? Как вы понимаете полиморфизм?
- Назовите основные принципы ООП. Сколько их всего?
- Как переопределить логику для метода? Какое слово нужно добавить для метода, чтобы у нас была возможность его переопределить?
- Что будет, если не переопределить метод класса?

21. Generic-методы, универсальность

- Что такое generic-метод? Как мы можем передавать тип данных в метод?
- В какой момент времени метод будет понимать, какой тип данных мы передали в метод?
- Могу ли я вернуть из метода переменную с типом данных, который я передала?
- Могу ли я сделать параметр с типом данных, который я передала? Как тогда изменится вызов этого метода?

22. Методы расширения, this

- Что такое метод расширения? Могу ли я просто так вызвать метод расширения? К чему он относится? Как с ним работать?
- Что такое this? На что указывает this? Приведите два основных примера, где стоит использовать слово this
- Как сказать методу расширения, с какими переменными он должен работать?
- Какие основные принципы создания методов расширений? Сколько их всего?