## Теоретическая часть

* Способы передачи параметров в методы C#:
  + по значению – позиционный
  + по значению – по именам параметров
  + по ссылке – ключевое слово **ref**, входные и выходные параметры
  + выходные параметры – ключевое слово **out**
* Передача переменного количества параметров в методы – ключевое слово **params**, псевдомассив параметров
* Классы в C# - начало. Конструкторы.
* Ключевые слова **this**, **base**.
* Cвойства, понятие об автосвойствах, полных свойствах, вычисляемых свойствах.
* Ключевые слова **get**, **set**. Контекстное ключевое слово **value**

## Практическая часть

Разработайте консольное приложение для решения следующих задач. Используйте возможности консоли по работе с цветом. Не используйте лямбда-выражения и методы расширения для массивов (отмечены стрелкой вниз в списке методов, выводимых IntelliSense) при решении задач. Каждый класс разместите в отдельном файле исходного кода (VS по умолчанию создает новый класс в отдельном файле исходного кода).

Рекомендую также создать класс **App**, в котором и реализовывать методы решения задачи – по аналогии с тем, что Вы делали в курсе «ООП на C++». Создавать и использовать объект класса App Вы будете в методе Main класса Program.

**Задача 1.** Это задание Вы выполняли в курсе «ООП C++», будет полезно выполнить его на C#. Создайте класс, описывающий усеченный конус **Conoid** (радиус верхнего основания, радиус нижнего основания, высота – тип полей **double**). Реализуйте полные свойства, вычисляемые свойства для [**площади**](https://www-formula.ru/2011-09-21-04-35-14) и [**объема**](https://www-formula.ru/2011-09-21-10-55-40) усеченного конуса, метод формирующий строковое представление конуса (ToString(), выводить только радиус и высоту), метод формирующий строку таблицы со сведениями о конусе (радиус, высота, площадь поверхности, объем). Разработайте компараторы для сортировки по возрастанию объемов, для сортировки по убыванию высот.

Создайте класс **ArrayConoid**, хранящий массив из усеченных конусов – объектов класса **Conoid** и название коллекции конусов.

Реализуйте методы класса **ArrayConoid**:

* заполнение массива данными конусов (не менее 12 элементов, используйте генератор случайных чисел, вводить с клавиатуры ничего не надо)
* вычисления суммарного объема конусов
* вычисления суммарной площади поверхности конусов
* вывод названия коллекции и массива конусов в консоль, в табличном виде
* вывод названия коллекции и массива в консоль в табличном виде: радиусы и высота усеченного конуса, площадь и объем с выделением цветом конуса/конусов с максимальной площадью, также выводить итоговую информацию – суммарный объем конусов, суммарную площадь поверхности конусов
* сортировка массива конусов по возрастанию объемов
* сортировка массива конусов по убыванию высот

Продемонстрируйте работу методов **ArrayConoid**, напоминаю – в массиве должно быть не менее 12 конусов.

**Задача 2.** Разработайте класс Персона со следующими **полными свойствами**:

* фамилия и инициалы имени и отчества (например, Семенов Р.О.)
* возраст в полных годах (целое число)
* рост в сантиметрах (целое число)
* вес в кг (вещественное число)
* название города проживания (строка)

Заполнить массив данными для не менее чем 12 персон. Рекомендую для наработки навыков кодирования не генерировать объекты массива, а создавать с использованием свойств. Для массива персон выполнить следующие обработки:

* Вывести массив персон в консоль.
* Вывести в консоль персону/персон с максимальным ростом.
* Вывести в консоль персону/персон с минимальным возрастом.
* Разработайте методы-компараторы для упорядочивания массива. Упорядочить массив и вывести его в консоль. Упорядочивать по:
  + по городу проживания
  + по убыванию веса
  + по возрастанию роста

## Дополнительно

Запись занятия можно скачать по [**этой** **ссылке**](https://cloud.mail.ru/public/W4Wp/fpdd6okJV). Материалы занятия – в этом же архиве.