Лабораторная работа 10. Обобщения (generics). Часть 1.
https://metanit.com/sharp/tutorial/3.12.php

1. Создать и протестировать на подходящих примерах обобщенные методы

a) Обобщенный метод, меняющий местами два элемента массива с заданными индексами. Протестировать для массивов с элементами целого, вещественного, строкового типа, а также одного из ранее созданных собственных классов (Person, Animal,…).

b) Обобщенный метод, который принимает одномерный массив и один целочисленный индекс и возвращает массив, в котором элемент удален (массив на 1 элемент короче исходного, элементом под заданным индексом исключен из него)

c) Обобщенный метод, который принимает одномерный массив и возвращает его максимальный элемент. Использовать ограничение IComparable<T> для параметра типа

2. Реализовать интерфейсы IComparable<>, IComparer<>
https://metanit.com/sharp/tutorial/3.23.php

a) Для класса Person (поля класса: Имя, Фамилия, Дата рождения, Пол) реализовать обобщенный интерфейс **IComparable** <>,
при этом сравнение объектов Person проводить по полю Имя, в лексикографическом порядке.

Протестировать работу метода CompareTo()

- на двух отдельных объектах
- при сортировке массива объектов библиотечным методом
- на примере создания и использования собственного статического метода, который возвращает максимальный объект из полученного в аргументах массива

b) Для класса Person, дополнительно к интерфейсу **IComparable** <>,
создать два дополнительных класса компараторов, реализующие **IComparer** <> и выполняющие

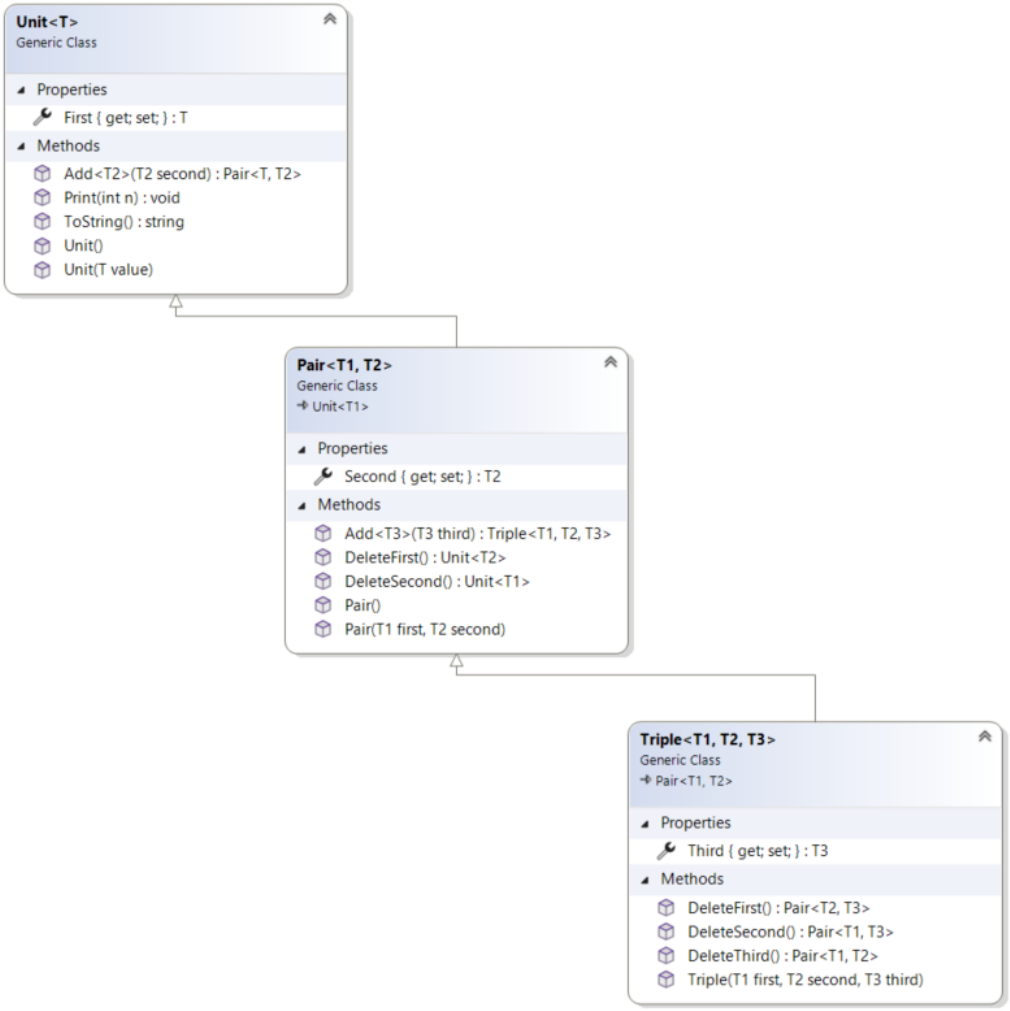
- сравнение по Возрасту
- сравнение по значению поля Пол (по числовым кодам символов в поле Пол)

Протестировать работу метода Compare()

- при сортировке массива объектов методом Array.Sort(…, …)
- на примере использования собственного статического метода,
  который возвращает максимальный объект из полученного массива
- на примере собственного статического метода,
  который сортирует массив методом вставок

3. **Создать систему обобщенных классов для работы с кортежами:**

- Unit<A> (1 element)
- Pair<A,B> (2 elements)
- Triplet<A,B,C> (3 elements)



В каждом классе должны быть открытые свойства для хранения разнотипных элементов кортежа; конструкторы; метод для добавления элемента, возвращающий следующий в иерархии кортеж; метод для удаления элемента, возвращающий предыдущий в иерархии кортеж; метод Print для каждого объекта кортежа выводит все его элементы на экран n раз.

Протестировать на подходящих примерах.

4. Создайте собственную обобщенную реализацию стека на основе массива. Готовые реализации из Generics не использовать.
Использовать этот стек для решения следующих задач:

```
a) Ввести строку, которая содержит формулу вида:

формула = цифра | M(формула,формула) | m(формула,формула)

где цифра = 0|1|2|3|4|5|6|7|8|9

M – вычисление максимума
m – вычисление минимума

знак | - одна из альтернатив (или)

Вычислить числовое значение этой формулы. Например, значение формулы M(m(3,5),M(1,2))  равно 3.

b) В текстовом файле записана информация о людях (фамилия, имя, дата рождения, пол через пробел).
Вывести на экран вначале информацию о людях младше 40 лет, а затем информацию о всех остальных
Для обеих возрастных групп данные должны выводиться в порядке обратном их нахождению в файле, т.е.
последние в файле выводятся на экран раньше.
```