

Задание 2.

Реализовать класс, предоставляющий три метода расширения для обобщённого интерфейса *IEnumerable<T>*:

- генерация всех возможных сочетаний из *n* (кол-во элементов перечисления) по *k* (с точностью до порядка, элементы могут повторяться) из элементов входного перечисления:

Входное перечисление: [1, 2, 3]; *k* == 2

Выходное перечисление: [[1, 1], [1, 2], [1, 3], [2, 2], [2, 3], [3, 3]]

- генерация всех возможных подмножеств (без повторений) из элементов входного перечисления:

Входное перечисление: [1, 2]

Выходное перечисление: [[], [1], [2], [1, 2]]

- генерация всех возможных перестановок (без повторений) из элементов входного перечисления:

Входное перечисление: [1, 2, 3]

Выходное перечисление: [[1, 2, 3], [1, 3, 2], [2, 1, 3], [2, 3, 1], [3, 1, 2], [3, 2, 1]]

Для каждого из методов требуется проверка элементов входного перечисления на предмет попарного неравенства по отношению эквивалентности, передаваемому в метод в виде реализации обобщённого интерфейса *IEqualityComparer<T>* (если нашлись два равных по переданному отношению эквивалентности элемента, то должна быть сгенерирована исключительная ситуация типа *ArgumentException* для входного перечисления). Продемонстрировать работу реализованных методов.