Семинар 3.

Задачи для самостоятельного решения.

№1. Опишите класс **Fraction** для работы с дробями. Разработайте оконное приложение – калькулятор дробей. Дополните класс перегруженными операциями ++ и --, позволяющими добавлять к дроби единицу и вычитать из дроби единицу, соответственно. Калькулятор позволяет выполнять основные арифметические операции над дробями, а также преобразует простые дроби в десятичные и наоборот.

№2. Написать класс **Interference** со строковым свойством типа интерференции (фонетический, грамматический, лексический, орфографический) и полем, представляющим её мощность. Переопределить оператор **сложения**, сложить два объекта этого типа, затем **сериализовать** любым изученным способом и открыть в другом проекте. Вывести значения **десериализованного** объекта, а также оповестить пользователя об успешности сериализации. В случае сложения объектов с разным значением типа интерференции выбрасывать **исключение**.

№3. Описать класс отрезков числовой прямой **Interval**. Определить в нём:

* конструктор, принимающий концы отрезка (должен корректно обрабатывать случаи, когда левый конец больше правого);
* копирующий конструктор;
* метод **Length**, возвращающий длину отрезка;
* операции интервальной арифметики;
* переопределённый унаследованный **ToString** ().

Операции интервальной арифметики определяются следующим образом:

Предусмотреть возможные исключительные ситуации, если это необходимо.

Написать программу, использующую этот класс.

№4. Описать класс **Polynomial** квадратных многочленов вида . Определить в нем:

• конструктор, принимающий коэффициенты многочлена;

• копирующий конструктор;

• метод **Value**, возвращающий значение многочлена в заданной точке;

• операции сложения и вычитания;

• операции умножения и деления на действительное число;

• операцию вычисления остатка от деления одного многочлена на другой;

• переопределённый унаследованный **ToString** ().

Предусмотреть возможные исключительные ситуации, если это необходимо.

Написать программу, использующую этот класс.

№5. Описать класс **Money** денежных сумм, заданных в виде количества рублей и копеек. Определить в нем:

• конструктор, принимающий количество рублей и копеек (должен корректно обрабатывать случаи, когда копеек больше 100, или количества рублей и копеек имеют разные знаки);

• копирующий конструктор;

• метод **TransFerCost**, принимающий величину комиссии за денежный перевод в процентах и возвращающий его полную стоимость с точностью до копеек (например, для суммы 10 р. 15 к. и величины комиссии 5 % полная стоимость составляет 10 р. 66 к.);

• операции сложения и вычитания;

• операции умножения и деления на действительное число (результат должен округляться до копеек);

• переопределённый унаследованный **ToString** ().

Предусмотреть возможные исключительные ситуации, если это необходимо.

Написать программу, использующую этот класс.