

Лабораторная работа 3

Создайте класс «[Рациональное число](#)». Класс должен содержать два поля — числитель (типа `int`) и знаменатель (типа `unsigned int`).

Сделайте следующее:

- 1) Реализуйте функции (члены класса) с именами `GetNumerator()` и `GetDenominator()`, возвращающие числитель и знаменатель дроби соответственно.
- 2) Реализуйте функцию (член класса) с именем `Reduce()`, которая сокращает дробь (то есть делит числитель и знаменатель на их наибольший общий делитель). Вычисление наибольшего общего делителя двух чисел реализуйте в виде глобальной функции `gcd()` (используйте [алгоритм Евклида нахождения НОД](#)).
- 3) Реализуйте функции (члены класса) `GetReducedNumerator()` и `GetReducedDenominator()`, которые возвращают числитель и знаменатель в несократимом представлении соответственно.
- 4) Перегрузите унарные операторы `+` и `-`, операторы `+=`, `-=`, `*=`, `/=` и бинарные операторы `+`, `-`, `*`, `/` для операций с рациональными числами.
- 5) Перегрузите операторы сравнения `==`, `!=`, `<`, `>`, `<=`, `>=` для сравнения рациональных чисел между собой.
- 6) Перегрузите префиксные и постфиксные операторы `++` и `--`. Для увеличения и уменьшения рационального числа на единицу соответственно.
- 7) Перегрузите оператор `<<()` для вывода в поток объектов класса в человеко-читаемом виде.
- 8) Определите оператор преобразования объекта класса рациональное число к типу `double`.
- 9) В качестве отчета пришлите 3 файла: заголовочный файл (`rational.h`) с

описанием класса, файл (rational.cpp) с реализацией класса и всех необходимых функций, а также файл (main.cpp), в котором будут примеры, демонстрирующие возможности использования вашего класса.