# Задание по программированию: Тесты для класса Rational

Условие

Класс Rational представляет собой рациональное число и имеет следующий интерфейс

class Rational {

public:

  Rational();

  Rational(int numerator, int denominator);

  int Numerator() const;

  int Denominator() const;

};

Список требований, предъявляемых к реализации интерфейса класса Rational:

1. Конструктор по умолчанию должен создавать дробь с числителем 0 и знаменателем 1.
2. При конструировании объекта класса Rational с параметрами p и q должно выполняться сокращение дроби p/q.
3. Если дробь p/q отрицательная, то объект Rational(p, q) должен иметь отрицательный числитель и положительный знаменатель.
4. Если дробь p/q положительная, то объект Rational(p, q) должен иметь положительные числитель и знаменатель (обратите внимание на случай Rational(-2, -3)).
5. Если числитель дроби равен нулю, то знаменатель должен быть равен 1.

Разработайте набор юнит-тестов, которые будут проверять корректность реализации класса Rational. **Тестировать случай, когда знаменатель равен нулю, не надо.**

Начать работу вы можете с шаблона, который содержит наш фреймворк юнит-тестов и заготовку класса Rational.

[test\_rational.cpp](https://d3c33hcgiwev3.cloudfront.net/_de8ebd63ad7a6913565dbd0f6425ecdc_test_rational.cpp?Expires=1616976000&Signature=Y0ruwBCrD6qI8jlmnAFc8FrnwL~IsCbU-TEMe-GcCvBQjUdxqclwLsptqrrMhS7iPxeYuGAGTDN0avtRVHu7I0vB5hArCJ5coGhrLFiTUQAtv9X~GRk9SmyuRie5ZDfCj-7lQVNQF--0YGVV3EsJKeYManI17GCz-HIVikaXCXA_&Key-Pair-Id=APKAJLTNE6QMUY6HBC5A)

### Примечание

Перед отправкой вашего файла убедитесь, что он не содержит класса Rational. В противном случае вы получите ошибку компиляции. Наша тестирующая система умеет игнорировать функции в файлах (поэтому, например, в задаче "Тесты для функции IsPalindrom" необязательно удалять функцию IsPalindrom), но не умеет это делать для классов.