

[← Назад к неделе 4](#)[✕ Уроки](#)[Пред.](#)[Дальше](#)

Задание по программированию: Исключения в классе Rational

Вы не отправили работу. Для успешной сдачи вам необходимо набрать 1/1 баллов.

Срок сдачи Сдайте это задание до November 12, 11:59 PM PST

[Инструкции](#)[Моя работа](#)[Обсуждения](#)

Вспомним класс Rational, который мы реализовали в задачах, посвящённых перегрузке операторов. Там специально говорилось, что в тестах объекты класса Rational никогда не создаются с нулевым знаменателем и никогда не выполняется деление на ноль. Настало время избавиться от этого ограничения и научиться обрабатывать нулевой знаменатель и деление на ноль. В этой задаче вам предстоит это сделать.

Поменяйте реализацию конструктора класса Rational так, чтобы он выбрасывал исключение `invalid_argument`, если знаменатель равен нулю. Кроме того, переделайте реализацию оператора деления для класса Rational так, чтобы он выбрасывал исключение `domain_error`, если делитель равен нулю.

Напомним, что вам надо реализовать интерфейс

```
1 class Rational {
2 public:
3     Rational();
4     Rational(int numerator, int denominator);
5
6     int Numerator() const;
7     int Denominator() const;
8 };
```

и оператор деления для класса Rational

```
1 Rational operator / (const Rational& lhs, const Rational& rhs);
```

How to submit

When you're ready to submit, you can upload files for each part of the assignment on the "My submission" tab.

Так же, как и в задачах на перегрузку операторов, вам будет предложен файл с заготовкой кода. Функция `main` в этом файле содержит ряд юнит-тестов, которые позволят вам проверить свою реализацию. При проверке вашего решения оно будет тестироваться на более полном наборе тестов, так что успешное прохождение всех юнит-тестов не гарантирует, что ваша реализация будет зачтена.

Шаблон программы:

```
rational_throw.cpp
```

