

Задания по теме "Перегрузка операций. Функции-операции. Операции приведения типов. Друзья класса" в C++

Задание 1: Перегрузка арифметических операторов

Создайте класс `Fraction`, который представляет дробь (числитель и знаменатель). Перегрузите операторы `+`, `-`, `*`, `/` для выполнения арифметических операций с дробями. Убедитесь, что результат всегда приводится к наименьшему общему знаменателю.

Задание 2: Перегрузка оператора сравнения

Создайте класс `Student` с полями `name` (строка) и `grade` (оценка, `int`). Перегрузите оператор `==`, чтобы сравнивать студентов по их оценке. Также перегрузите оператор `<`, чтобы сравнивать студентов по алфавиту (по имени).

Задание 3: Перегрузка оператора ввода/вывода

Создайте класс `Book` с полями `title` (строка) и `year` (год издания, `int`). Перегрузите операторы `<<` и `>>` для ввода и вывода объектов класса `Book` в поток.

Задание 4: Перегрузка оператора инкремента

Создайте класс `Counter`, который представляет счетчик. Перегрузите операторы `++` (префиксный и постфиксный) для увеличения значения счетчика на 1.

Задание 5: Перегрузка оператора приведения типа

Создайте класс `Distance`, который хранит расстояние в метрах. Перегрузите оператор приведения типа `double`, чтобы объект класса `Distance` можно было неявно преобразовать в километры (деление на 1000).

Задание 6: Дружественная функция

Создайте класс `Rectangle` с приватными полями `width` и `height`. Напишите дружественную функцию `calculateArea`, которая вычисляет площадь прямоугольника и возвращает её.

Задание 7: Дружественный класс

Создайте класс `Secret` с приватным полем `data` (строка). Создайте дружественный класс `SecretAccess`, который имеет метод `printSecret`, выводящий значение `data`.

Задание 8: Перегрузка оператора присваивания

Создайте класс `DynamicArray`, который динамически управляет массивом целых чисел. Перегрузите оператор присваивания `=`, чтобы корректно копировать объекты этого класса (глубокая копия).

Задание 9: Перегрузка оператора индексации

Создайте класс `String`, который представляет строку. Перегрузите оператор `[]`, чтобы можно было получить доступ к символу строки по индексу. Убедитесь, что оператор работает как для чтения, так и для записи.

Задание 10: Комбинированная задача

Создайте класс `Matrix`, который представляет матрицу 2x2. Перегрузите операторы:

- `+` для сложения матриц.
 - `*` для умножения матриц.
 - `<<` для вывода матрицы в поток.
 - `[]` для доступа к элементам матрицы по индексам (например, `matrix(0, 1)`).
-