Министерство образования и науки Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

**«Пермский национальный исследовательский**

**политехнический университет»**

Кафедра «Информационные технологии и автоматизированные системы»

**О Т Ч Ё Т**

**по лабораторной работе**

Дисциплина: Основы алгоритмизации и программирования

Тема: Сохранение данных в файле с использованием потоков

Выполнил работу

студент группы РИС-22-1б

Рыжков Н.С.

Проверил

Доцент кафедры ИТАС

Полякова О.А.

Пермь, 2023

**Описание задачи**

Необходимо разработать класс Vector, который будет представлять собой динамический массив целых чисел. Класс должен содержать методы для доступа к элементам массива, а также перегруженные операторы для выполнения арифметических операций с векторами и целыми числами. Также необходимо разработать класс VectorIterator для обхода элементов вектора.

**Описание класса Vector**

Атрибуты

- `data` - вектор целых чисел, содержащий элементы вектора

Методы

- `Vector()` - конструктор по умолчанию, создающий пустой вектор

- `Vector(int size)` - конструктор, создающий вектор заданного размера

- `Vector(const Vector& other)` - конструктор копирования

- `Vector& operator=(const Vector& other)` - оператор присваивания

- `~Vector()` - деструктор

- `int& operator[](int index)` - оператор доступа к элементу вектора по индексу

- `const int& operator[](int index) const` - константный оператор доступа к элементу вектора по индексу

- `friend Vector operator+(const Vector& a, const Vector& b)` - оператор сложения векторов

- `friend Vector operator+(const Vector& v, int x)` - оператор сложения вектора и целого числа

- `VectorIterator begin()` - метод для получения итератора на начало вектора

- `VectorIterator end()` - метод для получения итератора на конец вектора

- `int size() const` - метод для получения размера вектора

**Описание класса VectorIterator**

Атрибуты

- `vec` - ссылка на вектор, который нужно обойти

- `index` - индекс текущего элемента вектора

Методы

- `VectorIterator(Vector& vec, int index)` - конструктор, создающий итератор для указанного вектора и индекса

- `VectorIterator& operator++()` - оператор инкремента, переводящий итератор на следующий элемент вектора

- `VectorIterator& operator--()` - оператор декремента, переводящий итератор на предыдущий элемент вектора

- `bool operator==(const VectorIterator& other) const` - оператор сравнения, проверяющий равенство текущего итератора с другим итератором

- `bool operator!=(const VectorIterator& other) const` - оператор сравнения, проверяющий неравенство текущего итератора с другим итератором

- `int& operator\*() const` - оператор доступа к текущему элементу вектора

**Заключение**

Предложенный класс Vector и класс VectorIterator позволят удобно работать с динамическими массивами целых чисел и обходить их элементы.

