ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»

Отчёт по лабораторной работе № 8

«Реализация класса. Вектор»

Выполнила работу

Сторожева Евгения

Академическая группа №C3100

Санкт-Петербург

2024

1. Введение

Заданием была реализации собственной структуры данных-класса

вектор. В ней должны присутствовать основные методы для работы класса: создание, сложение, вычитание, умножение, деление. Ограничения состояли в том, что как базовая структура данных используются динамические массивы int, есть необходимость реализовать проверку на совпадение размерности, а также ручное управление памятью

Реализация: описание решения и вывод программы

1. Реализация

Код, представленный на рисунках №1-4, реализует класс MyVector, представляющий динамический вектор. В классе определены приватные члены (int\* data - указатель на динамический массив и size\_t size - размер вектора), что обеспечивает контроль над доступом к данным.

В данном классе есть конструктор, который позволяет задать начальные значения членам класса, выделяет необходимое количество памяти для хранения элементов вектора и может проверять входные параметры и выбрасывать исключения, если они недопустимы.

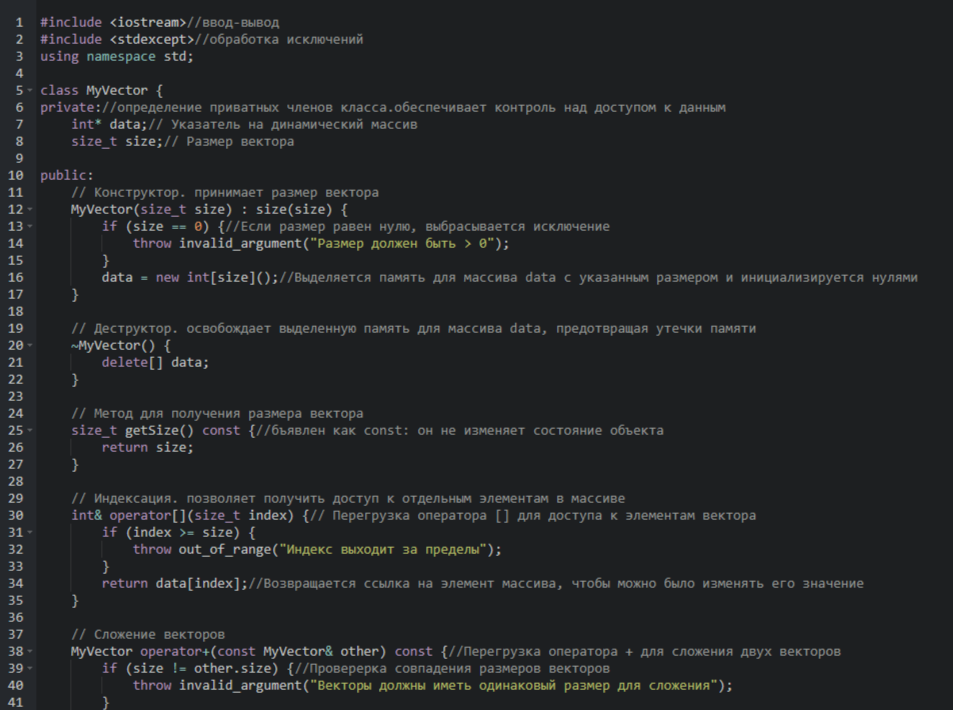
Деструктор в классе освобождает динамически выделенную память, предотвращая утечки. Он выполняется автоматически, когда объект класса выходит из области видимости или явно уничтожается.

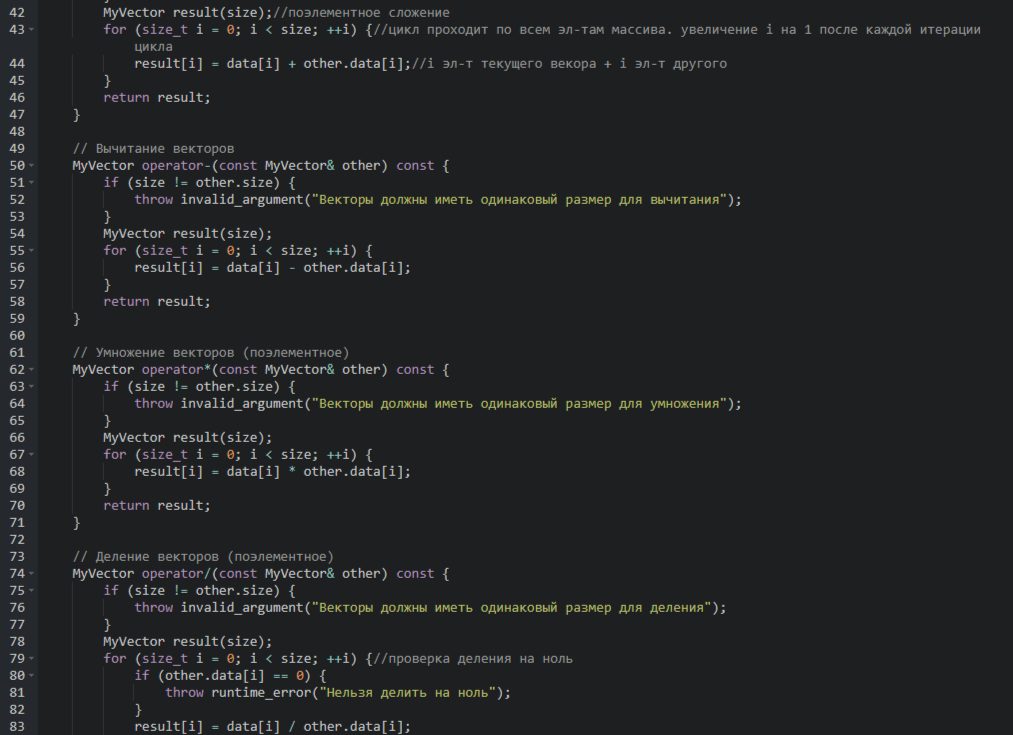
Так, ручное управление памятью осуществляется в data = new int[size](), где используется оператор new для выделения памяти под массив целых чисел и в операторе delete[] для освобождения ранее выделенной памяти.

В реализованной структуре данных есть все необходимые методы, упомянутые в описании задания: создание и операции сложения, вычитания, умножения, деления. В каждой операции проверяется совпадение размерности векторов и создается новый вектор result, заполненный поэлементным выполнением текущей операции.

В основной функции инициализируются два вектора и вывод программы, представленный на рисунке №5.

ПРИЛОЖЕНИЕ А

Рисунок№1-Реализация класса MyVector



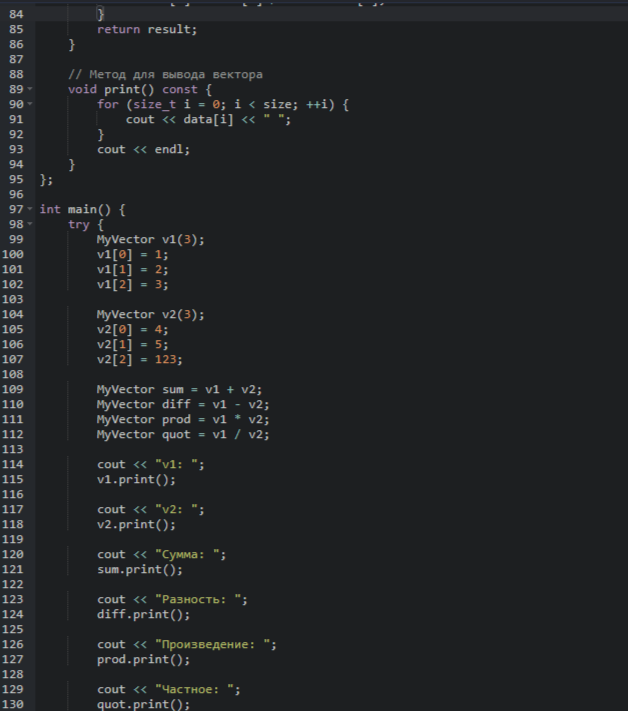
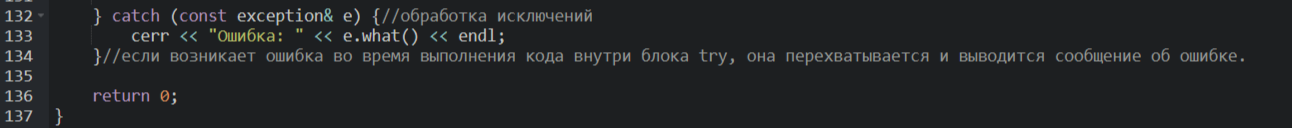
Рисунок №2-Реализация класса MyVector

Рисунок №3-Реализация класса MyVector

 Рисунок№4-Реализация класса MyVector

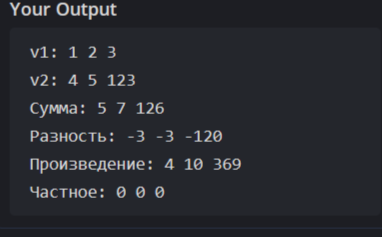


Рисунок №5-Вывод программы