1. Цель работы

Разработать программу на языке C++, реализующую классы для работы с векторами и матрицами, а также основные операции векторной алгебры:

- Сложение, вычитание, умножение векторов и матриц.
- Умножение на скаляр.
- Умножение матрицы на вектор.
- Контроль создания и уничтожения объектов с помощью статического счетчика.

2. Описание программы

2.1. Структура программы

Программа состоит из двух основных классов:

1. vect – класс для работы с векторами.
2. matr – класс для работы с матрицами.

Каждый класс содержит:

- Поля:
  - int dim – размерность вектора/матрицы.
  - double* b (для vect) / double* a (для matr) – указатель на массив элементов.
  - int num – уникальный номер объекта.
  - static int count – статический счетчик объектов.
- Конструкторы и деструкторы.
- Перегруженные операторы (+, -, *, = и др.).
- Методы доступа к элементам (set, get).

2.2. Основные алгоритмы

2.2.1. Класс vect (вектор)

- **Конструкторы:**
  - **По умолчанию** – создает пустой вектор.
  - **С параметром** dimension – выделяет память под вектор заданной размерности.
  - **Копирования** – создает копию существующего вектора.
- **Деструктор** – освобождает память.
- **Операторы:**
  - + (сложение векторов) – поэлементное сложение.
  - - (вычитание векторов) – поэлементное вычитание.
  - * (скалярное произведение) – сумма произведений соответствующих элементов.
  - - (унарный минус) – инвертирование знаков всех элементов.
  - = (присваивание) – копирование данных с проверкой на самоприсваивание.
- **Дополнительные методы:**
  - set(int index, double value) – установка значения элемента.
  - get(int index) – получение значения элемента.

2.2.2. Класс matr (матрица)

- **Конструкторы:**
  - **По умолчанию** – создает пустую матрицу.
  - **С параметром** dimension – выделяет память под квадратную матрицу.
  - **Копирования** – создает копию существующей матрицы.
- **Деструктор** – освобождает память.
- **Операторы:**
  - + (сложение матриц) – поэлементное сложение.
  - - (вычитание матриц) – поэлементное вычитание.
  - * (умножение матриц) – стандартный алгоритм умножения (строка × столбец).
  - * (умножение матрицы на вектор) – преобразование вектора по матрице.
  - - (унарный минус) – инвертирование знаков всех элементов.
  - = (присваивание) – копирование данных с проверкой на самоприсваивание.
- **Дополнительные методы:**
  - set(int row, int col, double value) – установка значения элемента.
  - get(int row, int col) – получение значения элемента.

3. Реализация программы

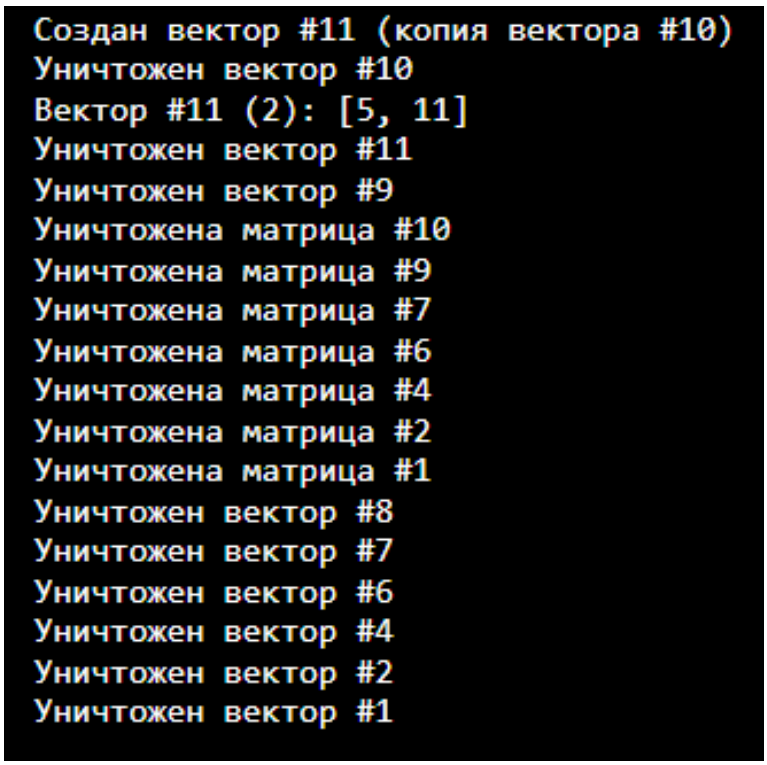
3.1. Основные функции

- print_vect(const vect& v) – вывод вектора на экран.
- print_matr(const matr& m) – вывод матрицы на экран.
- main() – демонстрация работы классов:
  - Создание векторов и матриц.
  - Тестирование всех операций.
  - Вывод результатов.

4. Тестирование программы

```
=== Работа с векторами ===
Создан вектор #1 размерности 3
Создан вектор #2 размерности 3
Вектор #1 (3): [1, 2, 3]
Вектор #2 (3): [4, 5, 6]
Сложение векторов #1 и #2
Создан вектор #3 размерности 3
Создан вектор #4 (копия вектора #3)
Уничтожен вектор #3
Вектор #4 (3): [5, 7, 9]
Вычитание векторов #1 и #2
Создан вектор #5 размерности 3
Создан вектор #6 (копия вектора #5)
Уничтожен вектор #5
Вектор #6 (3): [-3, -3, -3]
Унарный минус для вектора #1
Создан вектор #7 размерности 3
Вектор #7 (3): [-1, -2, -3]
Скалярное произведение векторов #1 и #2
Скалярное произведение: 32
Умножение числа 2.5 на вектор #1
Создан вектор #8 размерности 3
Вектор #8 (3): [2.5, 5, 7.5]
```

```
=== Работа с матрицами ===
Создана матрица #1 размерности 2x2
Создана матрица #2 размерности 2x2
Матрица #1 (2x2):
[1, 2]
[3, 4]
Матрица #2 (2x2):
[5, 6]
[7, 8]
Сложение матриц #1 и #2
Создана матрица #3 размерности 2x2
Создана матрица #4 (копия матрицы #3)
Уничтожена матрица #3
Матрица #4 (2x2):
[6, 8]
[10, 12]
Вычитание матриц #1 и #2
Создана матрица #5 размерности 2x2
Создана матрица #6 (копия матрицы #5)
Уничтожена матрица #5
Матрица #6 (2x2):
[-4, -4]
[-4, -4]
Унарный минус для матрицы #1
Создана матрица #7 размерности 2x2
Матрица #7 (2x2):
[-1, -2]
[-3, -4]
Умножение матриц #1 и #2
Создана матрица #8 размерности 2x2
Создана матрица #9 (копия матрицы #8)
Уничтожена матрица #8
Матрица #9 (2x2):
[19, 22]
[43, 50]
Умножение числа 3 на матрицу #1
Создана матрица #10 размерности 2x2
Матрица #10 (2x2):
[3, 6]
[9, 12]
Создан вектор #9 размерности 2
Умножение матрицы #1 на вектор #9
Создан вектор #10 размерности 2
Создан вектор #11 (копия вектора #10)
Уничтожен вектор #10
```



5. Вывод

В ходе лабораторной работы:

- Разработаны классы vect и matr для работы с векторами и матрицами.
- Реализованы основные операции векторной алгебры.
- Обеспечен контроль создания и уничтожения объектов через статический счетчик.
- Проведено тестирование всех операций.

Программа корректно выполняет все требуемые операции и выводит информацию о работе объектов.