

Модуль 2. Лабораторные работы

Задача 1

Создать шаблонный вариант функций задачи 3 (Модуль 1).
Продemonстрировать работу для типов *int* и *double*.

Задача 2

Создать класс **CVector**.

Функциональные возможности:

- Конструкторы:
 - инициализация (размер и данные). По умолчанию – нулевой размер;
 - копия;
- Функции:
 - копирование ({размер и данные}, {объект} (учесть ситуацию присваивания самому себе));
 - вывод данных в стандартный поток по формату:

Num: {size} Data: {x1} {x2}...

Продemonстрировать работу методов с выводом результата.

Задача 3

Класс **CVector**.

- Добавить в класс функции (возможно использование функций Модуля 1):
- скалярное произведение двух векторов (внешняя friend-функция);
 - норма;
 - сумма и разность двух векторов (как внешние friend-функции);
 - прибавление к вектору, вычитание из вектора.

Продemonстрировать работу методов с выводом результата.

Задача 4

Класс **CVector**.

- Добавить в класс операторы:
- оператор присваивания (учесть ситуацию присваивания самому себе);
 - оператор индексации (l-value и r-value);
 - скалярное произведение (внешний friend-оператор);
 - сумма, разность (как внешние friend-операторы);
 - прибавление к вектору, вычитание из вектора;
 - норма (как функтор).

Продemonстрировать работу методов с выводом результата.

Задача 5

Создать производные классы **CVector2** и **CVector3** на базе класса **CVector** с обычной (**out()**) и виртуальной (**vout()**) перегрузкой функции вывода данных, соответственно.

Формат вывода **vout()**:

Num: {size}

1: {x1}

2: {x2}

...

Продemonстрировать работу функции вывода для каждого производного класса **CVector2** и **CVector3**:

- производного класса;
- указателя на базовый класс от производного;
- ссылки на базовый класс от производного.

Задача 6

Класс **CVector**.

Добавить методы:

- Конструктор:
 - копия перемещением;
- Функции:
 - копирование перемещением;
 - оператор присваивания перемещением.

Продemonстрировать работу методов с выводом результата.

Задача 7

Создать шаблон-класс **TVector** (параметры шаблона - размер и тип данных).
Функциональные возможности соответствуют классу **CVector**.

Продemonстрировать работу функций и операторов с выводом результата:

- инициализация (копия и данные);
- арифметические операции;
- скалярное произведение;
- норма.