**Задачи для подготовки к зачету по дисциплине «Программирование на С++».**

**№1.**

1. Написать класс, реализующий динамический массив вещественных элементов. Обеспечить безопасность работы с динамической памятью. Написать конструктор с двумя параметрами для создания и заполнения массива (по какой-нибудь формуле, использующей второй параметр; если переданы некорректные параметры, то указатель на динамический массив должен быть направлен на **nullptr**, а размер массива обнулен). Написать константный метод для вывода массива на экран. Копирование, присваивание, перемещение для объектов этого класса запретить.

2. Для класса из п.1 написать класс-наследник, не имеющий дополнительных (своих) данных, но имеющий перегруженный оператор “ **[ ]** ” для обращения к элементам массива.

3. Для класса из п.2 ввести возможность отслеживать количество существующих объектов при помощи статических членов класса.

Продемонстрировать работоспособность написанного кода.

Предусмотреть дружественный интерфейс.

Запрещается использование библиотек, за исключением <iostream> .

Предполагаемое время выполнения: 60 мин.

**№2.**

1. Написать класс, реализующий динамический массив целочисленных элементов. Обеспечить безопасность работы с динамической памятью. Написать конструктор с двумя параметрами для создания и заполнения массива (по какой-нибудь формуле, использующей второй параметр; если переданы некорректные параметры, то указатель на динамический массив должен быть направлен на **nullptr**, а размер массива обнулен). Написать константный метод для вывода массива на экран. Копирование, присваивание, перемещение для объектов этого класса запретить.

2. Для класса из п.1 реализовать возможность копирования, присваивания, перемещения.

3. Внутри функции **main ( )** объявить динамический массив на **N** объектов класса (из п.1 и п.2). Заполнить объекты массивами разных размеров и значений посредством цикла. Написать функцию циклического сдвига объектов в массиве. Размер сдвига должен передаваться в качестве параметра. Замена объектов в массиве должна осуществляться перемещением (допускается использовать функции **move ( )** и **swap ( )** ).

Продемонстрировать работоспособность написанного кода.

Предусмотреть дружественный интерфейс.

Запрещается использование библиотек, за исключением <iostream> и <utility>.

Предполагаемое время выполнения: 70 мин.

**№3.**

1. Написать класс, реализующий двумерный динамический массив вещественных элементов. Обеспечить безопасность работы с динамической памятью. Написать метод (с тремя параметрами) для создания и заполнения массива (по какой-нибудь формуле, использующей третий параметр; если переданы некорректные параметры, то указатель на динамический массив должен быть направлен на **nullptr**, а размер массива обнулен). Написать константный метод для вывода массива на экран. Написать константный метод, рассчитывающий и возвращающий сумму динамического массива (если массив не задан, то возвращает ноль). Копирование, присваивание, перемещение для объектов этого класса запретить.

2. Для класса из п.1 реализовать возможность копирования, присваивания, перемещения.

3. Внутри функции **main ( )** объявить динамический массив на **N** объектов класса (из п.1 и п.2). Заполнить объекты матрицами разных размеров и значений посредством цикла. Написать функцию сортировки массива объектов по признаку суммы элементов матрицы, осуществляющую замену объектов перемещением (допускается использовать функции **move ( )** и **swap ( )** ).

Продемонстрировать работоспособность написанного кода.

Предусмотреть дружественный интерфейс.

Запрещается использование библиотек, за исключением <iostream> и <utility>.

Предполагаемое время выполнения: 80 мин.