**Задание на лабораторную работу по дисциплине**

**«Объектно-ориентированное программирование»**

**Задача №1.**

Создать класс **CMatrix**, в котором объявить одномерный массив. Массив проинициализировать случайными числами. В данном классе создать метод, позволяющий преобразовать массив так, чтобы первый элемент поменялся местами с последним, второй элемент – с предпоследним элементом и т.д. Проверить работоспособность созданного класса в методе Main(). На экран вывести исходный и преобразованный массивы. Тип проекта - консольное приложение.

**Задача №2.**

Создать класс с именем PC (компьютер), содержащий следующие поля:

* название;
* модель компьютера;
* дата выпуска, параметры: частота, объем памяти.

Создать коллекцию для хранения экземпляров созданного класса (не менее 5 элементов).

Написать программу, выполняющую следующие действия:

1. вывод всех элементов коллекции;
2. добавление элемента в коллекцию с клавиатуры;
3. удаление элемента из коллекции.
4. упорядочивание по убыванию объема памяти;
5. вывод на экран информации о компьютере, модель которого введена с клавиатуры (если такого нет, вывести соответствующее сообщение).

Тип проекта - Windows приложение.

**Задача №3.**

Разработать приложение, моделирующее процесс заполнения информацией файлового сервера. Создать класс, в котором определить закрытое поле - объем памяти файлового сервера (для доступа к данному полю реализовать соответствующее свойство), методы - проверка файла на ошибки (объем памяти равен 0 или превышает 700 мб), копирование файла на сервер, удаление с сервера заданного объема информации. В методе Main() проверить работоспособность созданного класса. Тип проекта - Windows приложение.

**Задача №4.**

Предметная область: **Банк**. Система хранит информацию о вкладчиках и сделанных ими вкладах. Класс «**вкладчик**» содержит имя вкладчика и величину вклада. Некоторым вкладчикам при создании вклада на счет может дополнительно перечисляться фиксированная сумма. В классе «**банк**» реализовать методы добавления нового вкладчика и метод вычисления общей суммы вкладов.

**Задача №5.**

Построить иерархию классов – **персона, студент, преподаватель**. В базовом классе реализовать виртуальный метод. В производных классах переопределить виртуальный метод. В методе Main() проверить работоспособность созданной иерархии классов и переопределенных методов. Тип проекта - консольное приложение.

**Задача №6.**

Создать классы: **Human** (человек), **Earth** (Земля), **Train** (поезд) и интерфейс: **IMove** с методом Move(int deltaT) и свойством Current. Метод Move() моделирует движение с шагом deltaT, свойство Current – показывает текущее положение. Реализовать в классах наследование от интерфейса IMove. Проверить работу классов в методе Main(). Логику движения моделируемых сущностей придумать самостоятельно.

**Задача №\***

Реализовать системы электрических источников и приборов, соединенных между собой через шнуры. В интерфейсах должна быть предусмотрена возможность получения информации о напряжении и максимальной мощности, которую поддерживает элемент. Прибор должен иметь наименование, потребляемую мощность, а источник и провод – списки (массив) подключенных приборов.

***Интерфейсы:***

* IElectricSource (источник тока)
* IElectricAppliance (электрический прибор)
* IElectricWire (электрический шнур)

***Классы:***

* SolarBattery (солнечная батарея)
* DieselGenerator (дизельный генератор)
* NuclearPowerPlant (атомная электростанция)
* Kettle (чайник)
* Lathe (токарный станок)
* Refrigerator (холодильник)
* ElectricPowerStrip (электрический удлинитель)
* HighLine (высоковольтная линия)
* StepDownTransformer (понижающий трансформатор, должен реализовывать интерфейсы и потребителя и источника тока).