**Задание 1**.

Предприятие оказывает клиентам услуги по снабжению энергией. Расчёты за потреблённую энергию осуществляются ежемесячно. Клиенты предприятия делятся на **четыре** вида, каждый из которых имеет свой алгоритм вычисления стоимости потреблённой энергии:

1. Обычные – стоимость потреблённой энергии прямо пропорциональна потреблённой энергии. Одна единица потреблённой энергии стоит 15 коп.
2. С лимитом – первые **150** единиц энергии клиент получает по цене, как для обычных клиентов (15 коп.), а всё, что потреблено сверх **150** единиц, оплачивается по тарифу **на ⅓** больше (20 коп.)
3. Льготные №1 – стоимость потреблённой энергии прямо пропорциональна потреблённой энергии, но применяется льготный тариф, который составляет **⅔** от тарифа для обычных клиентов (10 коп.)
4. Льготные №2 – клиенты получают первые **50** единиц энергии бесплатно, а всё, что потреблено сверх этих **50** единиц оплачивают по тарифу как для обычных клиентов (15 коп.).

Разработать **консольную** программу, в которой создать массив из данных о потреблённой клиентами энергией **за** **один** месяц. В массиве должно быть данные о минимум трёх клиентах каждого вида. Объём потреблённой каждым клиентом энергии должен быть введён в программу с использованием константных значений (программа не должна запрашивать ввод данных с консоли).

Выполнить следующую обработку хранящихся в массиве данных и вывести результаты выполнения каждого шага на консоль:

1. Отсортировать массив по количеству потреблённой клиентами энергии по убыванию.
2. Отсортировать массив по величине оплаты клиентами по возрастанию.
3. Упорядочить массив по типу клиентов (первыми – обычные, затем – с лимитом, затем – льготные №1 и последними – льготные №2)
4. Вычислить общую сумму *SUM* оплаты всех клиентов за потреблённую энергию.
5. Вычислить общий размер льготы *LG=SUM0***–***SUM*. Где *SUM0* – общая сумма оплаты всех клиентов за потребленную энергию, если бы они все были обычными (без льгот и лимитов).

**Задание 2**

Создать классы квадратов и окружностей, реализующие общий интерфейс «Геометрические фигуры»:

интерфейс Геометрическая фигура

Геометрическая фигура

Квадрат

Круг

Интерфейс должен определять следующие элементы: свойство, возвращающее площадь фигуры, метод вывода информации, и наследовать интерфейс **IComparable** длясравнения фигур по типу.

Класс квадратов должен содержать следующие элементы: поле-сторону квадрата, конструктор, реализованные элементы интерфейса, метод вычисления периметра. Класс кругов должен содержать следующие элементы: поля - радиус, цвет фигуры, конструктор, реализованные элементы интерфейса.

Создать класс-контейнер для фигур, содержащий поле-массив фигур (ссылок интерфейсного типа), конструктор, методы для сортировки по площади и по типу фигуры, а также реализующий интерфейс **IEnumerable**.

Дополнительно создать класс , реализующий интерфейс **IComparer.** Использовать объект класса для сортировки фигур по площади.

В классе Program создать объект класса-контейнера с информацией о двух кругах и двух квадратах. Вывести информацию о фигурах, используя для просмотра содержимого контейнера оператор **foreach**. Сортировать информацию по типу. Сортировать информацию по площадям фигур.

**Задание 3**

Создать иерархию классов:

Человек

Студент

Класс «человек» должен содержать следующие элементы: поле-фамилия, поле-год рождения, поле статус (**студент, преподаватель, бизнесмен и т.д.**), конструктор с параметрами**,** конструктор без параметров, метод вывода информации об объекте, **виртуальный** методом **Info** возвращающим возраст. Использовать метод **Info** в методе вывода информации. Реализовать в классе «Человек» интерфейс **IComparable** длясравнения людей по возрасту.

Класс студентов должен содержать дополнительные поля-оценки по математике, физике и истории, конструктор с параметрами, метод для получения среднего балла за сессию. Переопределить метод **Info** в классе «Студент» так, чтобы он возвращал максимальную оценку.

Дополнительно создать класс , реализующий интерфейс **IComparer.** Использовать объект класса для сортировки списка людей по алфавиту.

В методе Main:

* формировать массив объектов класса человек (с клавиатуры не вводить, генератор случайных значений не использовать).
* Сортировать по возрасту.
* Сортировать по фамилии.
* Выводить информацию