**Практическая работа №19: Объектно-ориентированное программирование**

Выполните следующие задание:

В некой игре-стратегии есть солдаты и герои. У всех есть свойство, содержащее уникальный номер объекта, и свойство, в котором хранится принадлежность команде. У солдат есть метод "иду за героем", который в качестве аргумента принимает объект типа "герой". У героев есть метод увеличения собственного уровня.

В основной ветке программы создается по одному герою для каждой команды. В цикле генерируются объекты-солдаты. Их принадлежность команде определяется случайно. Солдаты разных команд добавляются в разные списки.

Измеряется длина списков солдат противоборствующих команд и выводится на экран. У героя, принадлежащего команде с более длинным списком, увеличивается уровень.

Отправьте одного из солдат первого героя следовать за ним. Выведите на экран идентификационные номера этих двух юнитов.

**ООП. Множественное наследование**

Выполните следующие задания:

***Задание № 1***

Задан класс Point, описывающий точку с координатами x, y на координатной плоскости. Используя механизм наследования, нужно расширить возможности класса Point путем добавления нового атрибута цвета. Для этого реализовать подкласс PointColor.

В классе Point реализовать следующие атрибуты:

* координаты точки;
* метод иницализации, который получает 2 параметра — координаты точки x, y;
* метод вычисления расстояния от точки до начала координат;
* метод getPoint(), который возвращает точку в виде списка.

В подклассе PointColor реализовать следующие атрибуты:

* цвет точки color;
* метод начальной инициализации, который получает 3 параметра: координаты точки и цвет;
* метод доступа к цвету color с именем getColor().

***Задание № 2***

Создать базовый класс «Домашнее животное» и производные классы «Собака», «Кошка», «Попугай», «Хомяк». С помощью конструктора установить имя каждого животного и его характеристики. Реализуйте для каждого из классов методы:

* Sound — издает звук животного (пишем текстом в консоль);
* Show — отображает имя животного;
* Type — отображает название его подвида.

***Задание № 3***

Создать базовый класс Employer (служащий) с функцией Print(). Она должна выводить информацию о служащем. В случае базового класса это может быть строка с надписью This is Employer class.

Создайте от него три производных класса: President, Manager, Worker.

Переопределите функцию Print() для вывода информации, соответствующей каждому типу служащего.