Рассмотрели применение разных списков. Для частого добавления в список разных объектов используется LinkedList, данный лист содержит отсортированную информацию с точной позиционностью, т.е. как мы туда положили объекты с такой позиции мы их можем достать. Для просмотра и выборки объектов из списка в основном используется ArrayList. Для построения очереди используют LinkedList, в котором уже реализованы большинство необходимых методов, таких как: peekFirst, peekLast…. etc.

Также в данном разделе было реализовано в нашей программе Tank уничтожение штаба. В своем коде переделал многие методы, перевел их на списки, в скорости работы не уверен, но программа начала смотреться приятней. Разработал механизм уничтожение любого списка объектов на поле танком, в нашем случае, танком агрессором. Были испробованы в данном методе сразу три типа списков: ArrayList, HashMap, LinkedMap. Map понадобились для хранения действий (Action) для разных уничтожаемых объектов на поле. Т.е. создал список с ключом «уничтожаемый объект» - «список действий для его уничтожения». Для объекта поля переписал метод hashCode и тут же столкнулся с проблемой содержания объектов в списке. По программе мы после уничтожения меняем свойство объекта, соответственно, меняется hashcode объекта, что приводит к невозможности найти данный объект в hashMap. Пришлось отказаться пока от переписывания метода hashCode, пока. Также убедился в невозможности предугадать сортировку в HashMap, необязательно если вы элемент положите первым, он там первым и будет. Тут думал, что глюк Idea, так как в отладчике все шло правильно, а при запуске через «зеленую стрелку», программа вела себя абсолютно по-другому.