1.  *Общее задание (****4 балла****)*. Написать функции основных алгоритмов  обработки односвязных списков. Написать программу,  тестирующую все функции. Реализовать следующие операции со односвязным списком:

a)       добавлении(включение) элемента начало/конец списка – addHead(), addTail();

b)      добавление/включение узла в указанную позицию связанного списка - insert();

c)       удаление(исключение) элемента из начала/конеца списка –  removeHead(),removeTail();

d)      удаление указанного узла из связанного списка remove();

e)       очистить список (удаление всех узлов) removeAll();

f)        вывод содержания связанного списка - outList ();

g)       поиск указателя узла по позиции и по значению data - findData(), findPos();

h)      проверка списка на пустоту isEmpty().

i)         извлечение (вернуть) значение поля данных  из указанного узла retrieve();

2.  *Задание по вариантам (****3 балла****).* Использовать функции операций над односвязным списком (см. п.1) для разработки функций следующих операций:

A.   Сцепление двух списков.

B.      Инвертирование списка, при котором первый элемент становится последним и т. д.

C.      Объединение двух упорядоченных списков в один упорядоченный список.

D.     Создание списка,  представляющего    собой    объединение   (по   операции  ИЛИ) элементов двух списков,

E.      Создание списка, содержащего    элементы,    общие    для    двух    других списков.

F.       Создание упорядоченного списка, путем вставки элементов на место заданного  порядка возрастания/убывания.

G.     Размещение элементов списка в возрастающем порядке.

H.     Вычисление суммы, произведения  вещественных значений элементов списка.

I.         Перемещение элемента node(p)  на n позиций вперед по списку.

J.        Создание копии списка.

3.  *Дополнительные задания (****1 балл****)*. Реализуйте функции итератора **curr**(текущий указатель) для доступа к элементам списка с операциями: 1) установка на начало списка,2) проверка конца списка,3) доступ к значению текущего элемента,4) переход к следующему элементу списка, 5) переход к предыдущему элементу списка (для списков на базе массива или двусвязных структур).