10810EECS204001  
Data Structures

Link List Practice

Submit to OJ: #12438

Description

In this practice, you are asked to implement some common operations using in a link list. There are total 10 operations to be done.

For each node, it contains two elements. One is its integer data and the other one is a pointer to next node.

**Operations:**

1. InsertBack data: insert a node with data to the back of the link list
2. InsertFront data: insert a node with data to the front of the link list
3. InsertAfter data ref: insert a node with data after the existed node with ref in the link list
4. InsertBefore data ref: insert a node with data before the existed node with ref in the link list
5. Delete data: delete the node with data (you may need to free the memory)
6. DeleteFront: delete the first node in the link list
7. DeleteBack: delete the last node in the link list
8. Reverse: reverse the nodes order from back to front
9. Rotate k: put the last k nodes into the front   
   (if k>total nodes, then k=k%total)
10. Swap data1 data2: swap the positions of node data1 and node data2  
    (if one node doesn’t exist, do nothing)
11. PrintChain: print the data from first node to last node in the form of   
    data1->data2. (we provide an unfinished code and PrintChain has already implemented)

Input

Each testcase will contain N operations. 0<N<100000

All integer data are between 0 to 100000.

There are no two nodes with the same data in this problem. That is, we will not have testcase like InsertFront 1, InsertFront 1.

Output

For output operation PrintChain, the form should be 1->2->3->4. Data and Data are separated by ‘->’.

Notice

The memory is limited, so beware with delete operations.