DAY 15

class Node {

int data;

Node next;

Node(int x) {

data = x;

next = null;

}

}

class LinkedList {

Node head;

/\*

Node createNode(int x) {

Node np = new Node();

np.data = x;

np.next = null;

return np;

}

\*/

Node createNode(int x) {

Node np = new Node(x);

return np;

}

void insertEnd(int x) {

Node nptr = createNode(x);

if(head == null) {

head = nptr;

}

else {

Node temp = head;

while(temp.next != null) {

temp = temp.next;

}

temp.next = nptr;

}

}

void insertBeg(int x) {

Node nptr = createNode(x);

if(head == null) {

head = nptr;

}

else {

nptr.next = head;

head = nptr;

}

}

Node searchNode(int x) {

if(head == null) {

return null;

}

Node temp = head;

while(temp != null) {

if(temp.data == x)

return temp;

temp = temp.next;

}

return null;

}

void insertAfter(int data, int x) {

Node nptr = createNode(data);

Node pos = searchNode(x);

if(pos == null) {

System.out.println("Node not found.");

return;

}

nptr.next = pos.next;

pos.next = nptr;

}

void deleteEnd() {

if(head == null) {

System.out.println("No node to delete");

return;

}

if(head.next == null) {

System.out.println("Deleted item: " + head.data);

head = null;

return;

}

Node temp = head;

while(temp.next.next != null) {

temp = temp.next;

}

System.out.println("Deleted item: " + temp.next.data);

temp.next = null;

}

void deleteBeg() {

if(head == null) {

System.out.println("No node to delete");

return;

}

System.out.println("Deleted item: " + head.data);

head = head.next;

}

void deleteAfter(int x) {

Node pos = searchNode(x);

if(pos == null) {

System.out.println("Node not found");

return;

}

if(pos.next == null) {

System.out.println("No node to delete after the given key");

return;

}

System.out.println("Deleted item: " + pos.next.data);

pos.next = pos.next.next;

}

void display() {

if(head == null) {

System.out.println("List Empty");

return;

}

System.out.println("List elements are: ");

Node temp = head;

while(temp != null) {

System.out.print(temp.data + " ");

temp = temp.next;

}

System.out.println();

}

int detectLoop() {

Node baby, big;

baby = big = head;

while(baby != null && big != null && big.next != null)

{

baby = baby.next;

big = big.next.next;

if(baby == big)

return 1;

}

return 0;

}

public static void main(String[] args) {

LinkedList li = new LinkedList();

li.head = null;

li.insertEnd(10);

li.display();

li.insertEnd(20);

li.display();

li.insertEnd(30);

li.display();

li.insertBeg(5);

li.display();

/\*

Node pos = li.searchNode(20);

if(pos == null)

System.out.println("Node not found");

else

System.out.println("Node found at " + pos);

\*/

li.insertAfter(25, 20); //5 10 20 25 30

li.display();

//li.deleteEnd();

//li.display();

//li.deleteBeg();

//li.display();

//li.deleteAfter(25);

//li.display();

Node last = li.searchNode(30);

Node pos = li.searchNode(20);

last.next = pos;

if(li.detectLoop() == 1)

System.out.println("Loop Detected");

else

System.out.println("No loop detected");

}

}