Algorithm Analysis and Data Structures: Assignment 1 – Linked Lists

Make a single linked list of integers.  There should be at least 15 nodes, the list should not be sorted.

Traverse the list.

Now sort the list.  The list should be sorted such that your program unlinks the nodes and relinks them so that they are sorted. (DO NOT SWAP THE VALUES IN THE NODES).

use selection sort.

Traverse the list again.

Code:

import java.util.\*;

public class LinkedList {

class Node{

int data;

Node next;

}

Node head;

public static void main(String[] args)

{

LinkedList l = new LinkedList();

Random r = new Random();

int value = 0;

for(int i = 1; i<=15; i ++)

{

value = r.nextInt(30);

l.AppendNode(value);

}

System.out.println("---Linked List before sorting---");

l.PrintLinkedList();

l.SortLinkedList();

System.out.println("---Linked List after sorting---");

l.PrintLinkedList();

}

//appendNode(val) - This method is used to add a new node if there is a head existing or creates a new head node to start the linked list with

public void AppendNode(int value)

{

if(head == null)

{

head = new Node();

head.data = value;

head.next = null;

}

else

{

Node newNode = new Node();

Node temp = head;

newNode.data = value;

newNode.next = null;

while(temp.next!= null)

temp = temp.next;

temp.next = newNode;

}

}

// PrintLinkedList() - this method traverses through the linked list and prints the nodes

public void PrintLinkedList()

{

if(head == null)

System.out.println("There are no elements in the linked list");

else

{

Node temp = head;

while(temp!=null)

{

System.out.println(temp.data);

temp= temp.next;

}

}

}

//SortLinkedList() - This method sorts the linked list using selection sort algorithm

public void SortLinkedList() {

Node temp=head;

Node pivot = head;

while(temp.next!=null) {

Node temp1 = temp;

Node minNode = temp;

Node minPrevious = temp;

Node minNext;

while(temp1.next != null ) {

if(minNode.data > temp1.next.data)

{

minNode = temp1.next;

minPrevious = temp1;

}

temp1 = temp1.next;

}

if(minNode != temp)

{

if(minPrevious != temp)

{

minPrevious.next = temp;

minNext = temp.next;

temp.next = minNode.next;

minNode.next = minNext;

}

else

{

temp.next = minNode.next;

minNode.next = temp;

}

if(pivot != temp)

pivot.next = minNode;

else

head = minNode;

}

temp = minNode;

pivot = temp;

temp = temp.next;

}

}

}

Instructions to compile:

1. Create a .java file in Sublime text or any text editor by the name LinkedList.java
2. Place the code in LinkedList.java and save
3. Open command prompt and redirect to the path where LinkedList.java is saved
4. Use javac LinkedList.java command, to compile the code
5. After successful compilation use java LinkedList command, to run the code and see the output

Screen Shot of the output

