

Software Development Fundamentals-II (15B11CI211)

Tutorial-7 [CO3]

Topic: Linked Lists

1. Write a function to get the intersection point of two Linked Lists.
2. Given a linked list and two keys in it, swap nodes for two given keys. Nodes should be swapped by changing links.

Input: 10->15->12->13->20->14, x = 12, y = 20

Output: 10->15->20->13->12->14

3. Write a program to reverse a linked list.
4. Write a program to reverse even elements in a Linked List.

Reversing the contiguous even elements will not take place when:

- a) The node's value is odd.
- b) The node's value is even but adjacent values are odd. In rest of the cases, the continuous block of even nodes can be reversed.

Initial list: 1 -> 2 -> 3 -> 3 -> 4 -> 6 -> 8 -> 5 -> NULL

Reversed list: 1 -> 2 -> 3 -> 3 -> 8 -> 6 -> 4 -> 5 -> NULL

5. Write a program to insert following elements into Singly Linked List (SLL). Further display the contents of SLL and count of unique elements (i.e. count of element 23 is 1 whereas for element 8, it is 3).

Elements of SLL: 23, 12, 8, 78, 5, 45, 8, 15, 18, 20, 2, 19, 9, 8, 25, 17

6. Write a program to move last m elements to the front of a given Linked List.
7. Write a program to add 1 to a number represented as linked list.

Number is represented in a linked list such that each digit corresponds to a node in linked list. Add 1 to it. For example 1999 is represented as (1->9->9->9) and adding 1 to it should change it to (2->0->0->0).