**SRM INSTITUTE OF SCIENCE AND TECHNOLOGY, Kattankulathur**





**School of Computing**

**21CSC201J – Data Structures and Algorithms**

**Topic: Introduction to Linked Lists**

**Activity: Fill in the blanks**

1. A linked list, in simple terms, is a **collection** of data elements.
2. The data elements in linked lists are called **nodes**.
3. Linked lists act as a **dynamic structure** to implement data structures such as stacks, queues, and their variations.
4. A linked list can be perceived as a **chain of nodes** in which each node contains one or more data fields and a pointer to the next node.
5. In a linked list, every node contains a pointer to another node which is of the same type, it is also called a **self-referential structure**.
6. Linked lists contain a pointer variable START that stores the **address of the first node** in the list.
7. We can traverse the entire list using **START** which contains the address of the first node; the next part of the first node in turn stores the address of its succeeding node.
8. If START = NULL, then the linked **list is empty**.
9. Linked lists provide an efficient way of storing **dynamic data** and perform basic operations such as insertion, deletion, and updation of information at the cost of extra space required for storing address of the next node.
10. In C, we can implement a linked list using the following code:

struct node

{

int data;

struct node \*next;

};