Data Structure Lab

Lab # 05

Doubly and Circular Linked List

Instructor: Muhammad Saood Sarwar

Email: saood.sarwar@nu.edu.pk

Course Code: CL2001

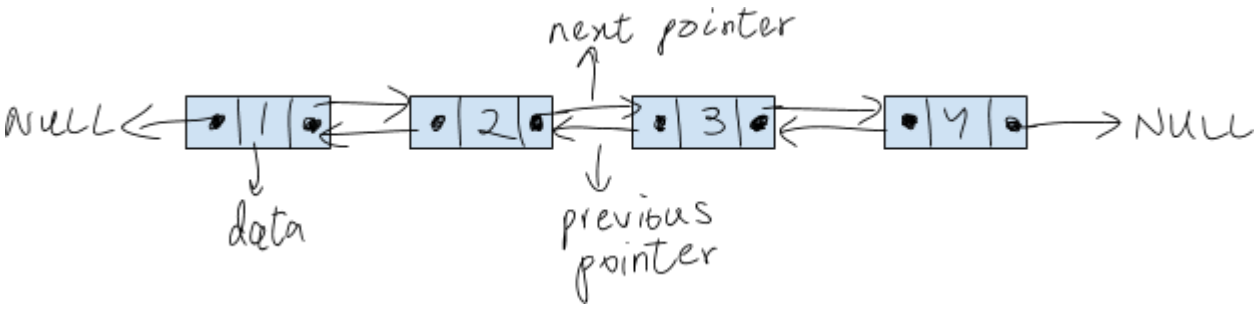
Department of Computer Science,

National University of Computer and Emerging Sciences FAST Peshawar Campus

Doubly Linked List:

```
class node
{
int info;          (data)
node *next;     (next pointer)
node *previous;     (previous pointer)
}
```

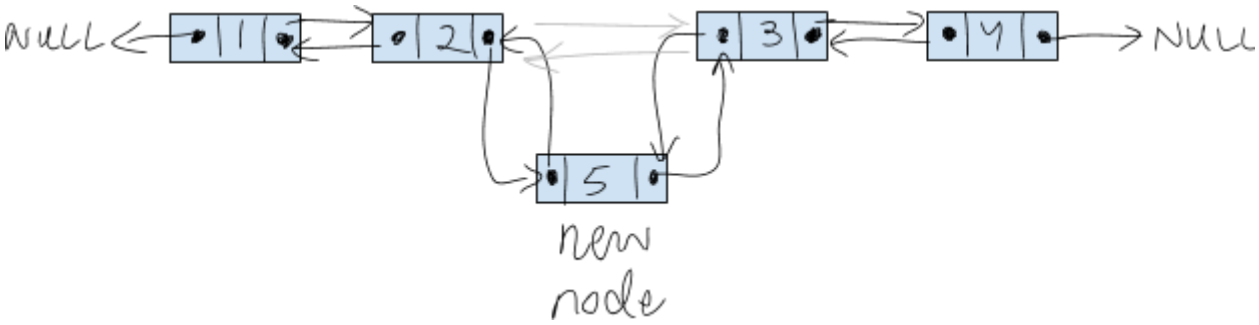
In Doubly Linked List, the list goes in both directions. You can move forward using ‘next’ pointer and move backward using ‘previous’ pointer.



Adding a new node to Doubly Linked List:

(grey lines show former link)

1. Create a new node containing data 5
2. Point 5’s next to 3
3. Point 5’s previous to 2
4. Point 3’s previous to 5
5. Point 2’s next to 5



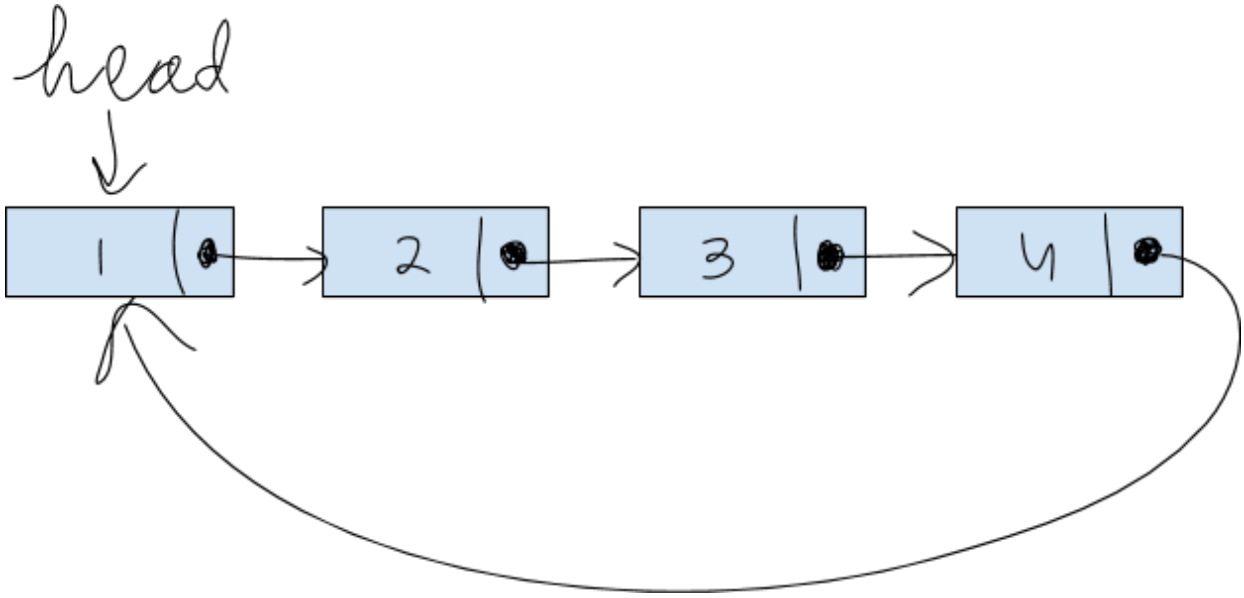
NOTE: Similarly you can add new node(s) at the end and start of Doubly but again, handle

Head Node with care.

NOTE (2): You can do Deletion as well. How? Ponder upon it.

Singly Ring/Circular List:

In Singly Ring List, the list doesn't end but goes on like a ring/circle. The next of last node points not to NULL but to the head of the list. This way a ring is formed.



Doubly Ring/Circular List:

In Doubly Ring, the list is formed in two directions i.e. forwards and backwards with last node's next pointing to head and head's previous pointing to last node.