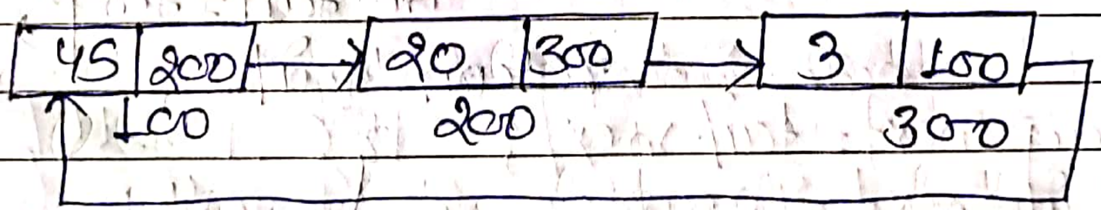
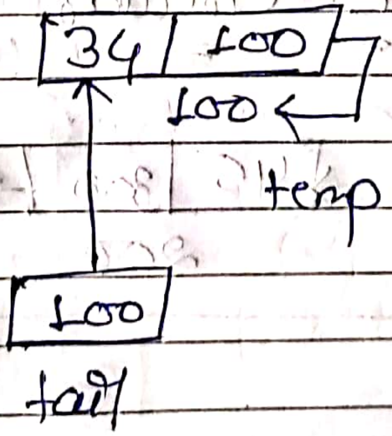
Circular linked list

① circular singly linked list

Circular singly linked list is similar to the singly linked list except that the last node of the circular singly linked list points to the first node.



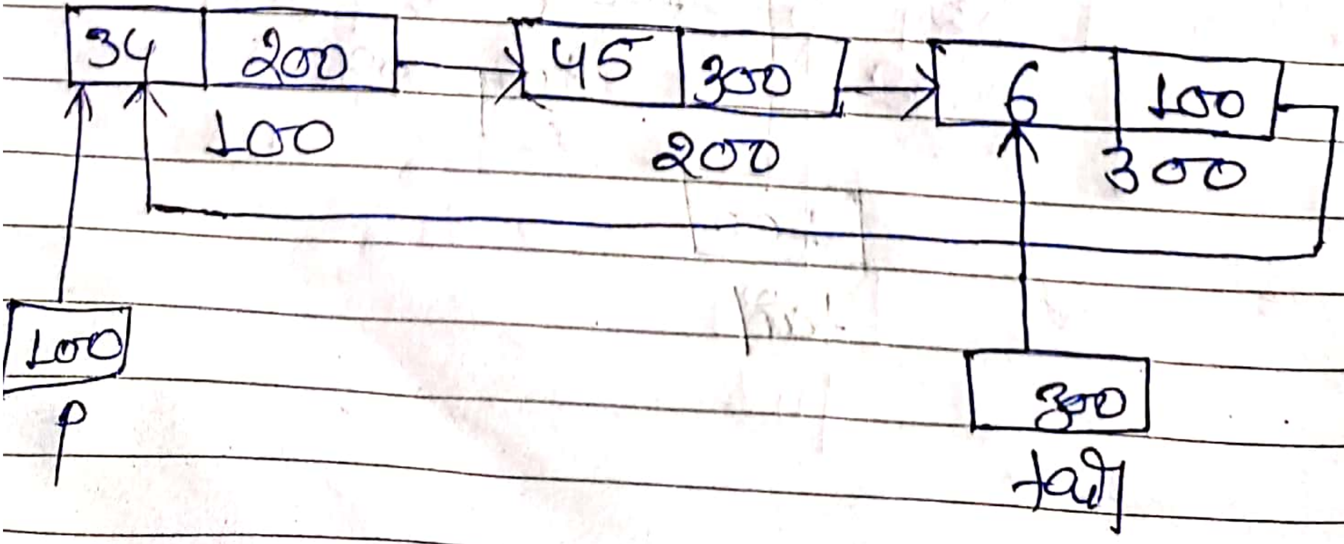
Circular singly linked list node.



creating a circular singly linked list node

```
node *circular_linked_list(node * tail, int data){
        node * temp = new node();
          temp → data = data;
           temp → next = temp;
              tail = temp;
        return tail;
     }
```

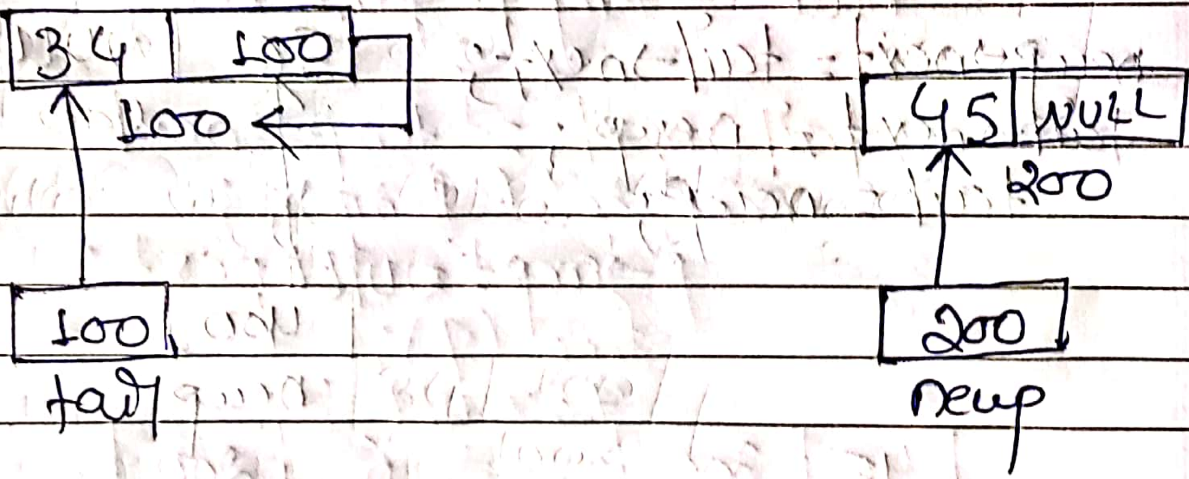
Traversing a circular singly linked list



```
node *p = tail->next;

  do{
  Cout << p->data << " ";
     p = p->next;
  } while ( p! = tail->next);
```

Insertion At the Beginning

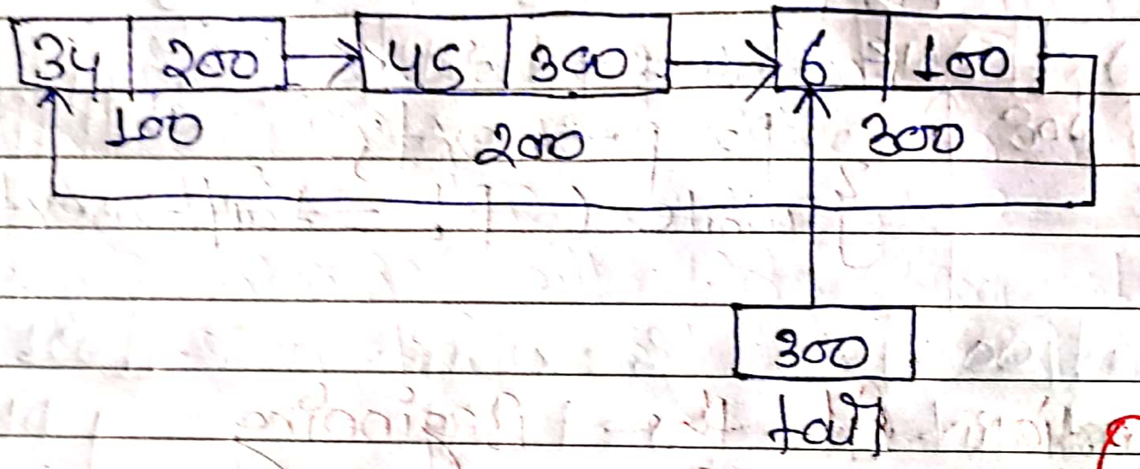


```
newp->next = tail->next;
tail->next = newp;
```

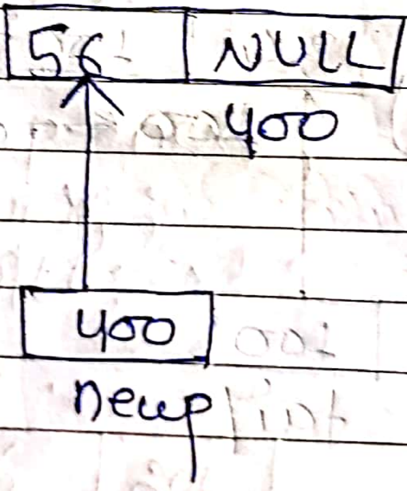
O(1)

Date ________
Page ________

Insertion at the end



O(1)

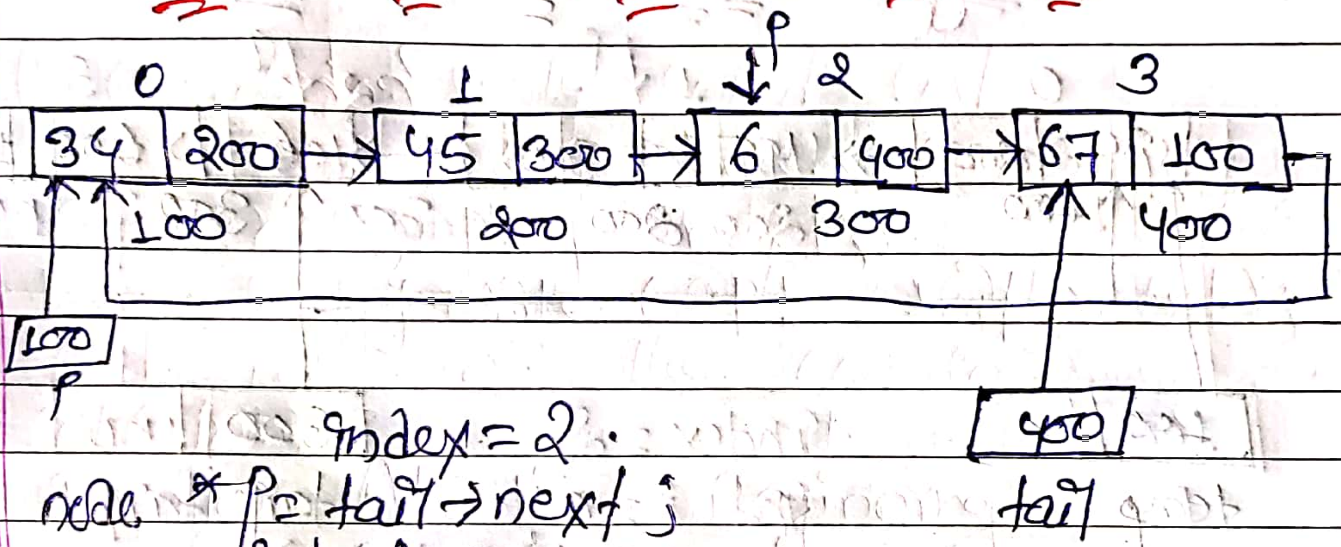


```
newp->next = tail->next;
tail->next = newp;
    tail = newp;
```

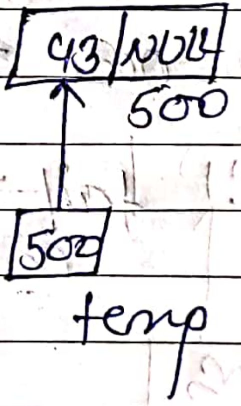
O(1)

Scanned with CamScanner

Insertion In Between the nodes



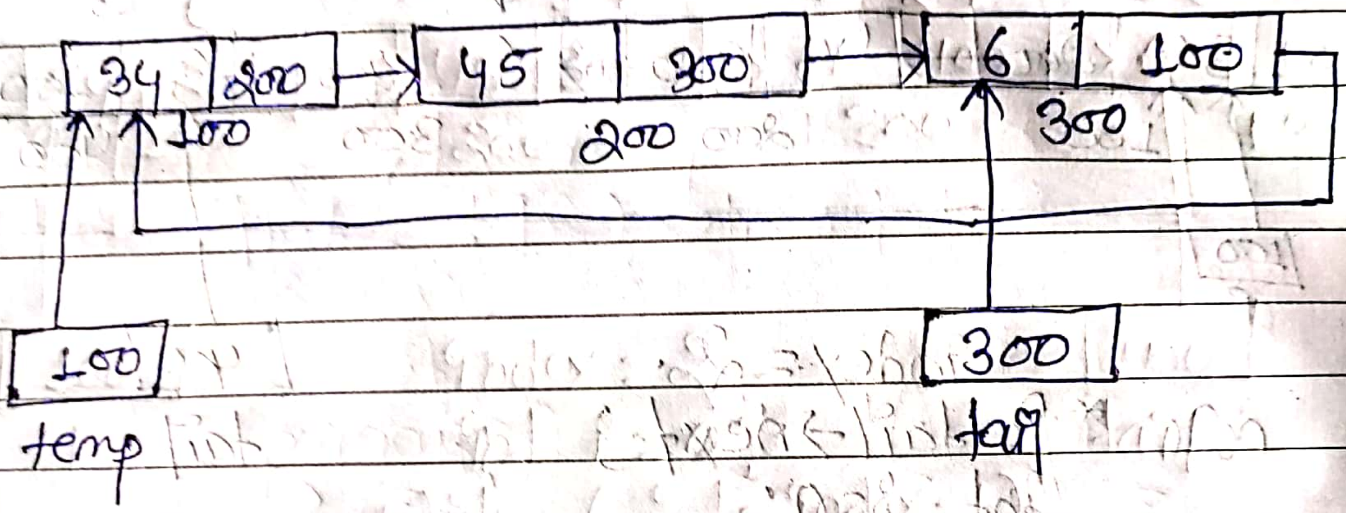
```
index = 2.
node *p = tail->next;
 int i=0;
```



```
while ( i != index ){
   p = p->next;
    i++;
   }
```

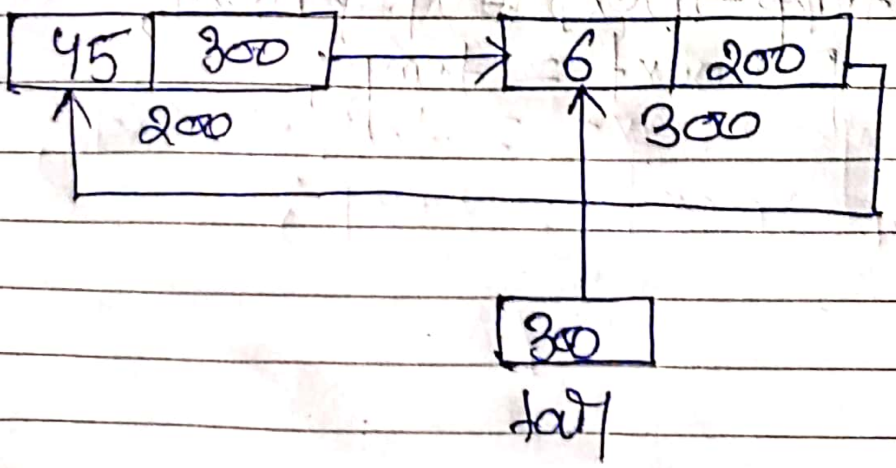
```
temp->next = p->next;
 p->next = temp;
```

Deleting first node

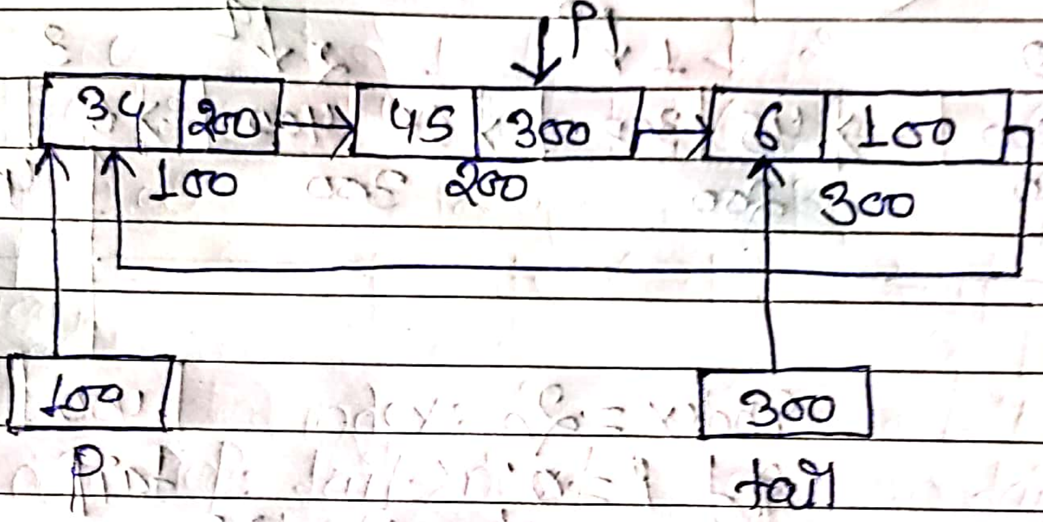


```
node * temp = tail->next;
tail->next = temp->next;
free ( temp);
temp = NULL;
```

O(1)



Deleting a last node

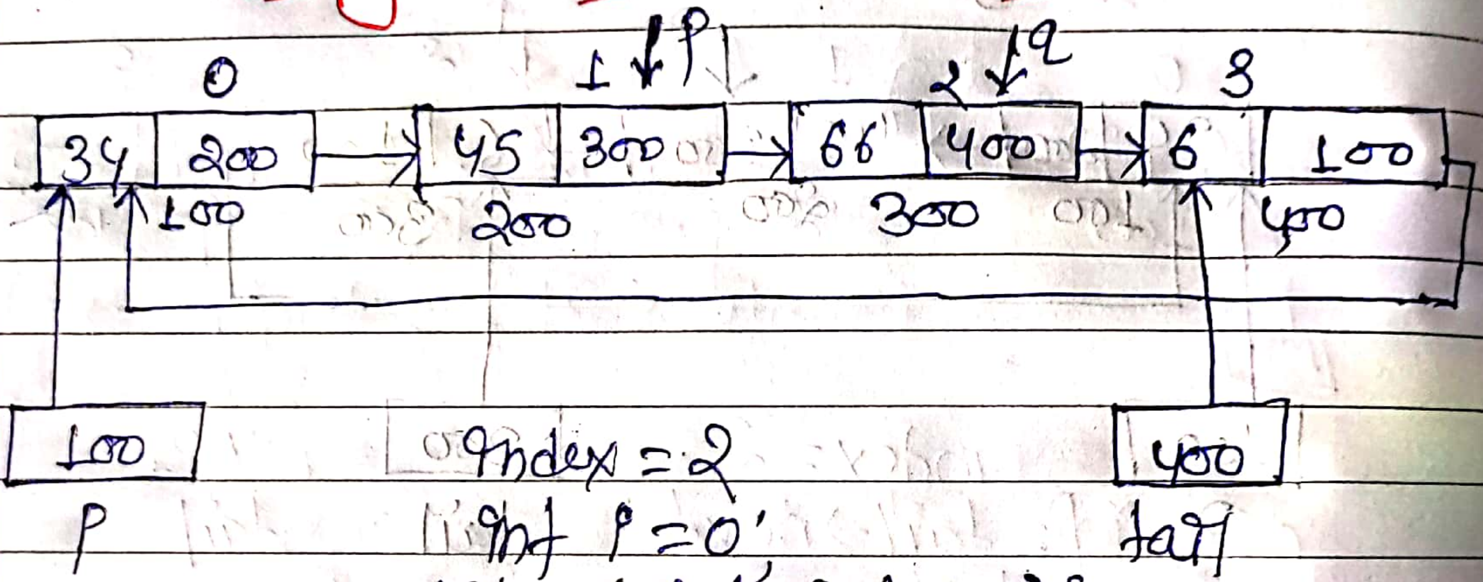


```
node *p = tail->next;
  while ( p->next != tail){
        p= p->next ;
      }

   p->next = tail->next;
     free (tail);
     tail = p;
```

Date
Page

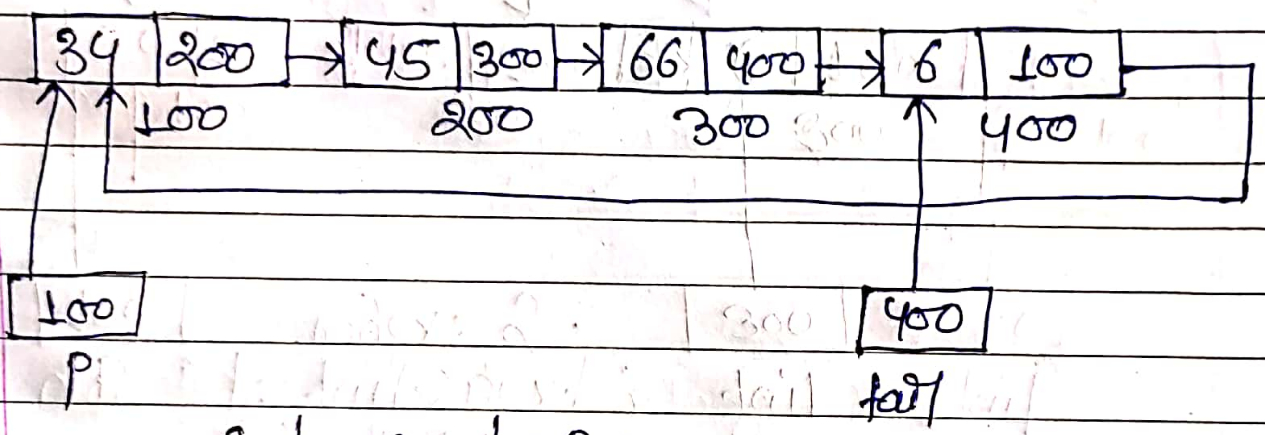
Deleting Intermediate node



```
index = 2
int i = 0;
while (i != index-1){
    p = p->next;
}

node *q = p->next;
p->next = q->next;
free (q);
q = NULL;
```

Counting the number of elements (node)



```
int count=0;
node * p= tail→next;

    if ( tail = =NULL) {
        cout<<" linked list is Empty";
        }
    else{
        do {
            count = count +1;
        cout<< p→ data "    ";
            p= p→next;
        }while ( p! =tail→next );
    cout<<"No. of node is :"<< count<<endl;
    cout << endl;
    }
```