1. Write a fu" insert_any()" for inserting a node at any given position of the linke list. Assume position starts at 0

Ans:- insert_any(int item, int key)

```
{
Struct node *new, *ptr;
ptr = start;
if (ptr->data != key)
{
ptr = ptr->link;
} else
{
```

```
new=(struct node *) malloc (sizeof(struct node));
new->link = ptr->link;
new->data = item;
ptr->link = new;
return;
}
}
```

2. Write a fun" "delete_beg()" for deleting a node from the begening of the linked list

Ans:- delete_beg()

```
{
struct node *ptr;
if (start==NULL)
{
printf("\n list empty!!");
return;
}
else
{
printf ("\n deleted element is: %d", *start);
ptr = start->link;
start->link = NULL;
start = ptr;
return;
}
```

3. Write a fn "delete_end()" for deleting a node from the end of the linked list

Ans:-

```
delete_end()
{
struct node *ptr, *ptr1;
ptr = start;
if (start == null)
 {
printf("\n list empty!!");
return;
}
else
{
while(ptr->link != null)
{
ptr1 = ptr;
ptr = ptr->link;
}
ptr1->link = null;
}
printf("\n deleted element is %d", *ptr);
}
```