1. Write a function “insert\_any()” for inserting a node at any given position of the linked list. Assume position starts at 0.
2. #include<stdio.h>

typedef int cat;

typedef struct node

{

int data;

struct node \*next;

}node;

node \*create(int x);

node \*insert\_any(node \*head, int x);

{

int main()

{

node \*head=NULL;

insert\_any(head,10);

}

node \*create(int x)

{

node \*new=(node \*)malloc(sizeof(node));

new -> data=x;

new -> next=NULL;

return new;

}

node \* insert\_any(node \*head,int x)

{

node \*new=create(x);

node \*temp=head;

if(!head)

{

head=new;

return head;

}

while(temp->next)

{

temp=temp->next;

}

temp->next=new;

return head;

}

node \*insert\_any(node \*n, int x,int pos)

{

node \*new=create(x)

new->node=temp;

new->data=x;

new->next=pos+1

if(temp->next==pos)

{

insert\_any(node \*n, int x,int pos)

}

2) Write a function “delete\_beg()” for deleting a node from the beginning of the linked list.

1. node \*delete\_beg(node \*\*new -> next, int x, int pos)

{

node \*temp= \*new->next, \*prev;

if(temp!=NULL && temp->data!=x)

{

\*new->next=temp->next;

free(temp);

return;

}

while(temp!=NULL && temp->data!=x)

{

prev=temp;

temp=temp->next;

}

if(temp==NULL) return;

prev->next=temp->next;

free(temp);

}

3)Write a function “delete\_end()” for deleting a node from the end of the linked list.

1. node \*delete\_end(node \*head)

{

if(head==NULL) return NULL;

if(head->next==NULL)

{

free(head);

return NULL;

}

node \*second\_last=head;

while(second\_last->next->next!=NULL)

second\_last=second\_last->next;

free(second\_last->next);

second\_last->next=NULL;

return head;

}