

2.WAP to Implement Singly Linked List with following operations

a) Create a linked list.

b) Insertion of a node at first position, at any position and at end of list.

Display the contents of the linked list

Shashank Patel C J

1BM22CS255

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
struct node
```

```
{
```

```
    int data;
```

```
    struct node* next;
```

```
};
```

```
void insertatbegin(struct node** head,int item)
```

```
{
```

```
    struct node* newnode=(struct node*)malloc(sizeof(struct node));
```

```
    newnode->next=*head;
```

```
    newnode->data=item;
```

```
    *head=newnode;
```

```
}
```

```
void insertatend(struct node** head,int item)
```

```
{
```

```
    struct node* newnode=(struct node*)malloc(sizeof(struct node));
```

```
    struct node* temp =*head;
```

```

newnode->next=NULL;

newnode->data=item;

if(*head==NULL)
{
    *head=newnode;

    return;
}

while(temp->next!=NULL)
{
    temp=temp->next;
}

temp->next=newnode;
}

void insertatspecific(struct node** head,int item,int loc)
{
    if(loc<=0)
    {
        printf("invalid position\n");

        return;
    }

    if(loc==1 || *head==NULL)
    {
        insertatbegin(head,item);

        return;
    }

    struct node* newnode=(struct node*)malloc(sizeof(struct node));

```

```

newnode->data=item;

struct node* temp =*head;

int count=1;

while(count<loc-1 && temp->next!=NULL)

{

    temp=temp->next;

    count++;

}

newnode->next=temp->next;

temp->next=newnode;

}

void display(struct node* head)

{

    struct node* temp=head;

    if(temp==NULL)

    {

        printf("linked list is empty\n");

        return;

    }

    while(temp!=NULL)

    {

        printf("%d ->",temp->data);

        temp=temp->next;

    }

    printf("NULL\n");

```

```
}
```

```
int main()
```

```
{
```

```
    struct node* head=NULL;
```

```
    insertatbegin(&head,10);
```

```
    insertatbegin(&head,20);
```

```
    insertatbegin(&head,30);
```

```
    insertatend(&head,40);
```

```
    insertatend(&head,50);
```

```
    insertatspecific(&head,25,2);
```

```
    insertatspecific(&head,35,4);
```

```
    display(head);
```

```
    return 0;
```

```
}
```

output:

```
30 ->20 ->10 ->NULL
30 ->20 ->10 ->40 ->50 ->NULL
30 ->25 ->20 ->35 ->10 ->40 ->50 ->NULL

Process returned 0 (0x0)   execution time : 0.031 s
Press any key to continue.
|
```