#include<stdio.h>

#include<conio.h>

typedef struct node

{

int data;

struct node \*link;

}node;

void insert\_at\_end(node \*\*,int);

void insert\_at\_beg(node \*\*,int);

void delete\_at\_end(node \*\*);

void delete\_at\_beg(node \*\*);

void print(node \*);

void main()

{

node \*start=NULL;

int ch,num;

clrscr();

while(1)

{

printf("\n 1.insert at beg\n 2.insert at end \n 3.delete at end \n4.delete at beg \n5.print\n 6. exit");

printf("\nenter your choice");

scanf("%d",&ch);

switch(ch)

{

case 1: printf("\nenter a num to insert");

scanf("%d", &num);

insert\_at\_beg(&start,num);

break;

case 2: printf("\nenter a num to insert");

scanf("%d", &num);

insert\_at\_end(&start,num);

break;

case 3: delete\_at\_end(&start);

break;

case 4: delete\_at\_beg(&start);

break;

case 5: print(start);

break;

case 6: exit(1);

}

}

}

void insert\_at\_end(node \*\*head,int num)

{

node \*temp,\*q;

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

temp->data=num;

temp->link=NULL;

if(\*head==NULL)

{

\*head=temp;

}

else

{

q=\*head;

while(q->link!=NULL)

{

q=q->link;

}

q->link=temp;

}

}

void insert\_at\_beg(node \*\*head,int num)

{

node \*temp, \*q;

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

temp->data=num;

temp->link=\*head;

\*head=temp;

}

void delete\_at\_end(node\*\* head)

{

node \*temp,\*q;

q=\*head;

if(\*head==NULL)

{

printf("\nlist is empty");

}

if(q->link==NULL)

{

\*head=NULL;

}

else

{

while(q->link->link!=NULL)

{

q=q->link;

}

}

temp=q->link;

q->link=NULL;

free(temp);

}

void delete\_at\_beg(node \*\*head)

{

node \*temp, \*q;

if(\*head==NULL)

{

printf("\n list is empty");

}

else

{

temp=\*head;

\*head=temp->link;

}

}

void print(node \*head)

{

node \*q;

q=head;

printf("\n");

if(head==NULL)

{

printf("list is empty");

return;

}

while(q!=NULL)

{

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

q=q->link;

}

}