Vidyalankar Polytechnic	DEPARTMENT OF COMPUTER ENGINEERING

Subject: - Data structure	**Subject Code:** 313301
Semester: - III	**Course:** COMPUTER ENGINEERING
Laboratory No: L003	**Name of Subject Teacher:** Prof. Imraan S.
Name of Student: Saad Sharif Kazi	**Roll Id: -** 24203A0013
Experiment No:	13
Title of Experiment	Write a C Program to Implement Singly Linked List with Operations: (i) Insert at end, (ii) Insert After, (iii) Delete (iv) Display

Aim: *Write a 'C' Program to Implement Singly Linked List with Operations: (i) Insert at end (ii) Insert after (iii)Delete(iv) Display

Algorithm:

Step 1: Start

Step 2: Define a structure Node with two fields:
data (integer)
next (pointer to next node)

Step 3: Define functions:
createlinkedlist(n) → creates a linked list with n nodes
createnode(data) → creates a new node with given data
printList(head) → prints all nodes in the list
end(head) → inserts a new node at the end of the list
mid(head) → inserts a new node at a given position
del(head) → deletes a node with a given data value

Step 4: In main program, declare head = NULL

Step 5: Accept the number of nodes n from the user

Step 6: Call createlinkedlist(n) to create a linked list with n nodes and store its address in head

Step 7: Display the linked list by calling printList(head)

Step 8: Call end(head) to insert a new node at the end of the linked list

Step 9: Display the updated list by calling printList(head)

Step 10: Call mid(head) to insert a new node at a specific position

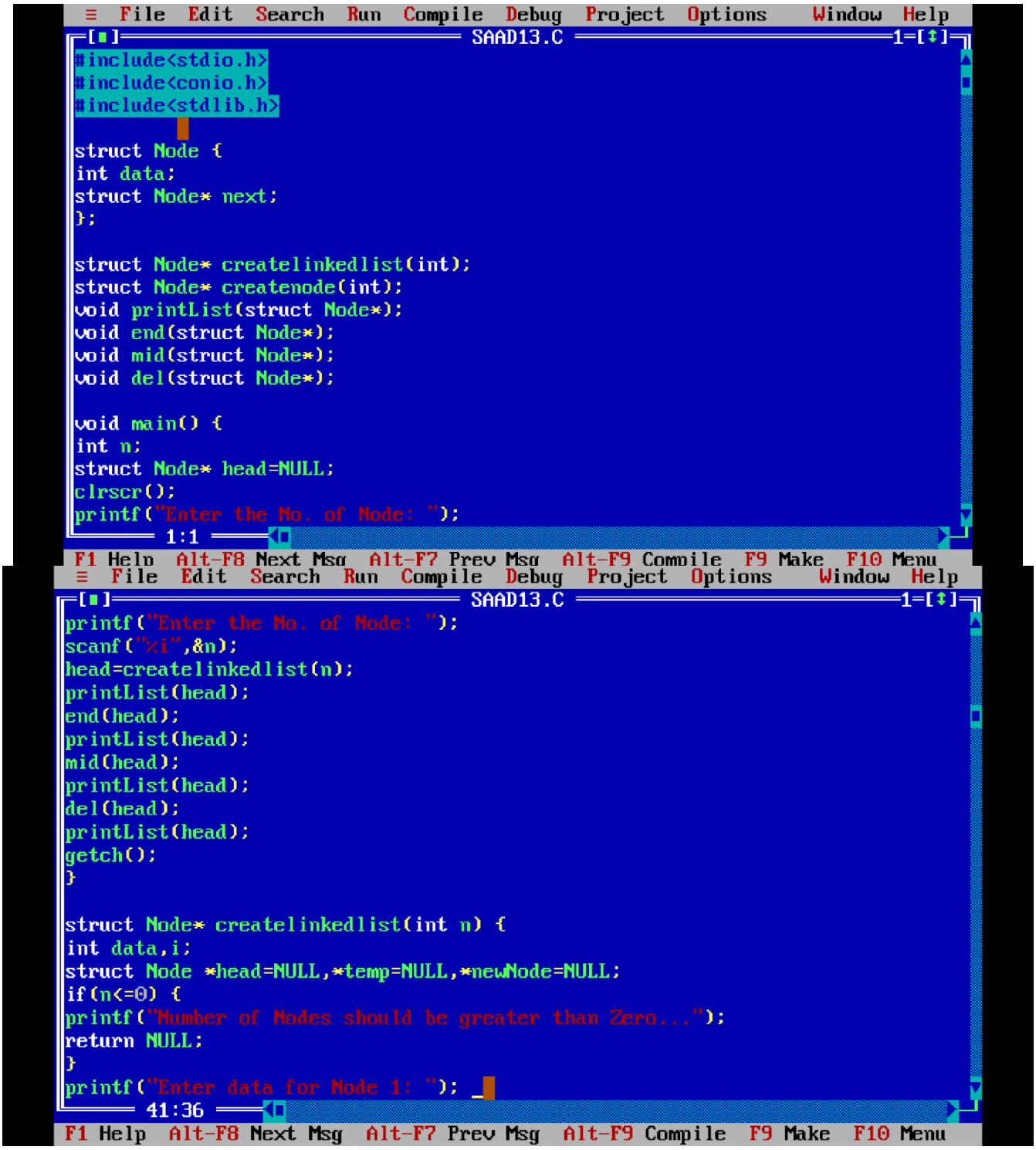
Step 11: Display the updated list by calling printList(head)

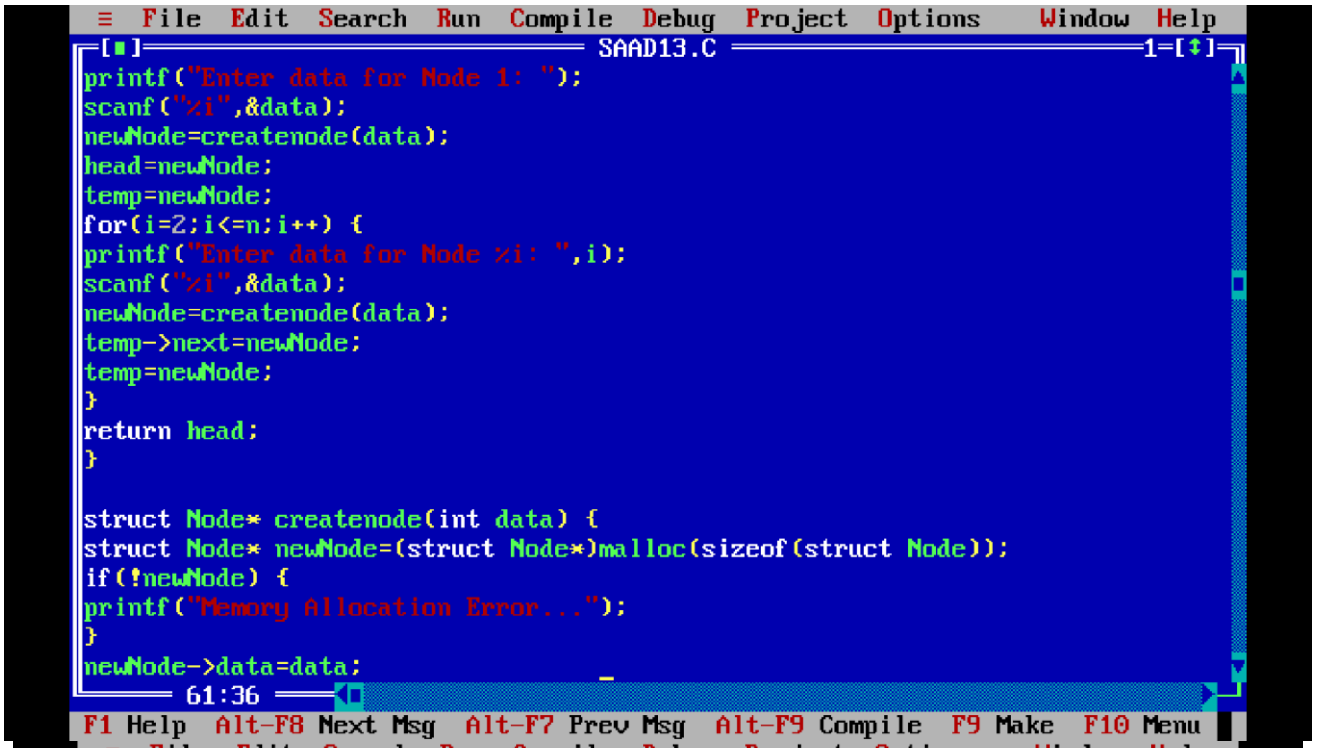
Step 12: Call del(head) to delete a node with a given data value

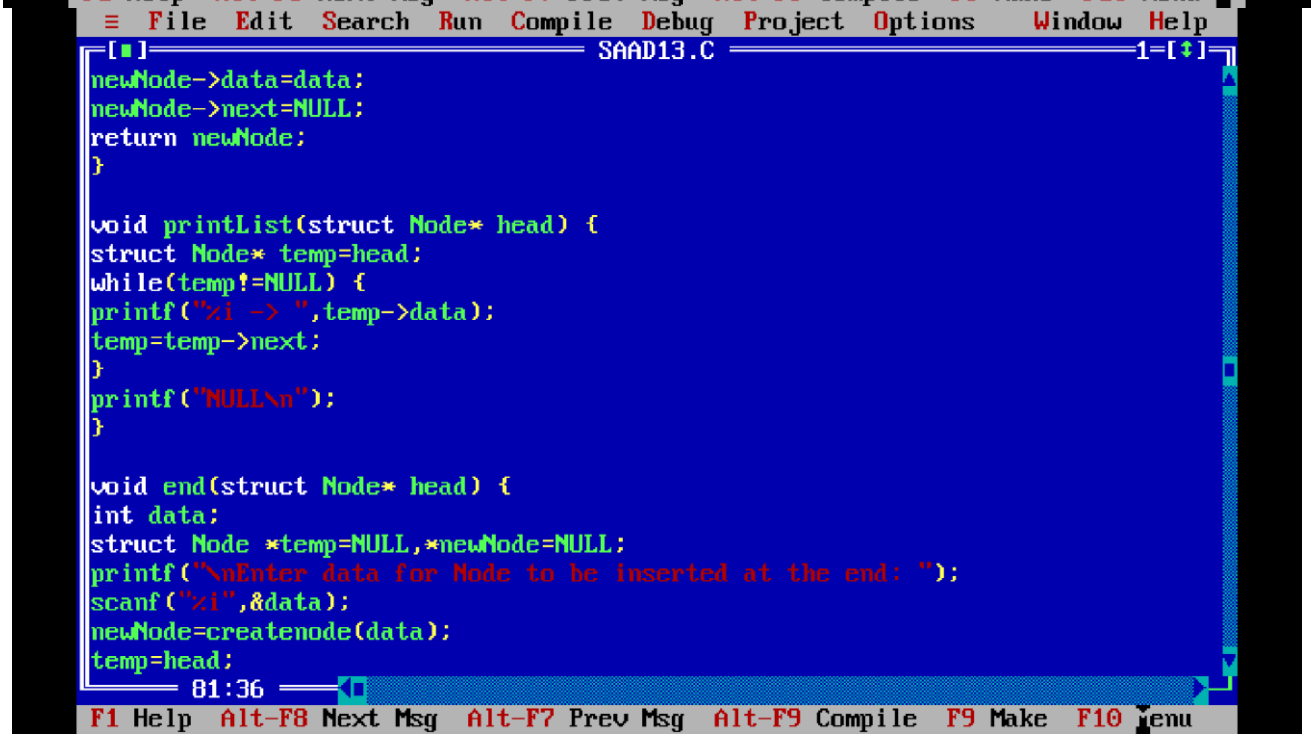
Step 13: Display the updated list by calling printList(head)

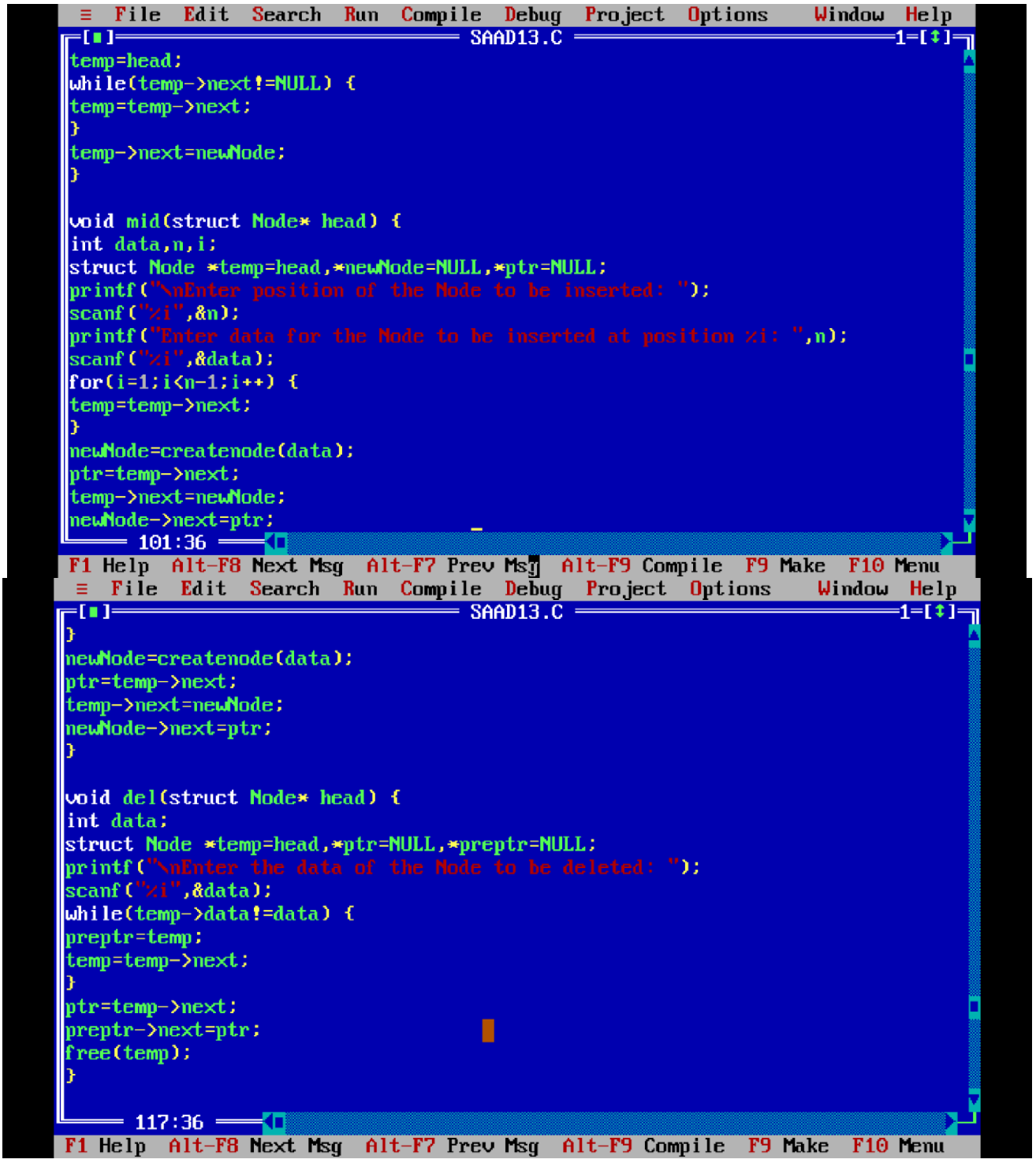
Step 14: Stop

CODE:

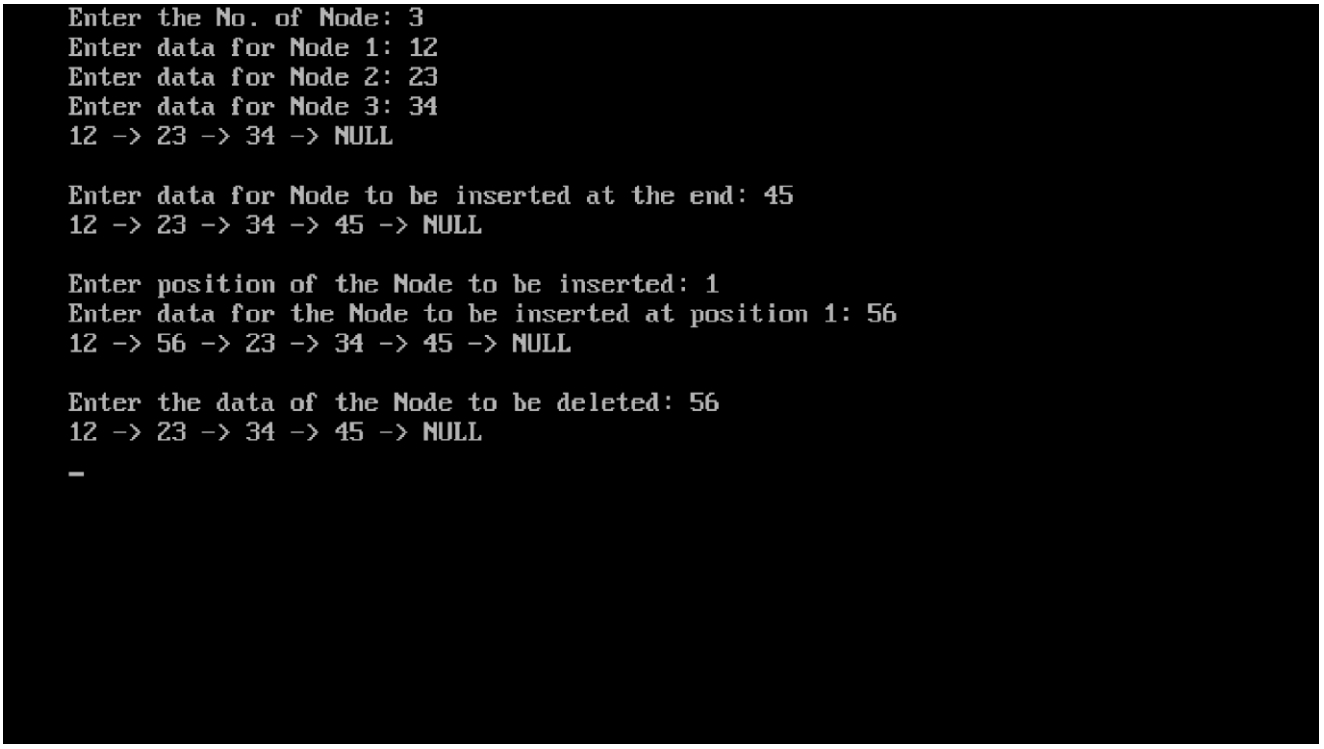

File Edit Search Run Compile Debug Project Options Window Help
SAAD13.C
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
struct Node {
int data;
struct Node* next;
};
struct Node* createlinkedlist(int);
struct Node* createnode(int);
void printList(struct Node*);
void end(struct Node*);
void mid(struct Node*);
void del(struct Node*);
void main() {
int n;
struct Node* head=NULL;
clrscr();
printf("Enter the No. of Node: ");
1:1
F1 Help Alt-F8 Next Msg Alt-F7 Prev Msg Alt-F9 Compile F9 Make F10 Menu
File Edit Search Run Compile Debug Project Options Window Help
SAAD13.C
printf("Enter the No. of Node: ");
scanf("%i",&n);
head=createlinkedlist(n);
printList(head);
end(head);
printList(head);
mid(head);
printList(head);
del(head);
printList(head);
getch();
}
struct Node* createlinkedlist(int n) {
int data,i;
struct Node *head=NULL,*temp=NULL,*newNode=NULL;
if(n<=0) {
printf("Number of Nodes should be greater than Zero...");
return NULL;
}
printf("Enter data for Node 1: ");
41:36
F1 Help Alt-F8 Next Msg Alt-F7 Prev Msg Alt-F9 Compile F9 Make F10 Menu


File Edit Search Run Compile Debug Project Options Window Help
SAAD13.C
printf("Enter data for Node 1: ");
scanf("%i",&data);
newNode=createnode(data);
head=newNode;
temp=newNode;
for(i=2;i<=n;i++) {
printf("Enter data for Node %i: ",i);
scanf("%i",&data);
newNode=createnode(data);
temp->next=newNode;
temp=newNode;
}
return head;
}
struct Node* createnode(int data) {
struct Node* newNode=(struct Node*)malloc(sizeof(struct Node));
if(!newNode) {
printf("Memory Allocation Error...");
}
newNode->data=data;
61:36
F1 Help Alt-F8 Next Msg Alt-F7 Prev Msg Alt-F9 Compile F9 Make F10 Menu


File Edit Search Run Compile Debug Project Options Window Help
SAAD13.C
newNode->data=data;
newNode->next=NULL;
return newNode;
}
void printList(struct Node* head) {
struct Node* temp=head;
while(temp!=NULL) {
printf("%i -> ",temp->data);
temp=temp->next;
}
printf("NULL\n");
}
void end(struct Node* head) {
int data;
struct Node *temp=NULL,*newNode=NULL;
printf("\nEnter data for Node to be inserted at the end: ");
scanf("%i",&data);
newNode=createnode(data);
temp=head;
81:36
F1 Help Alt-F8 Next Msg Alt-F7 Prev Msg Alt-F9 Compile F9 Make F10 Menu


File Edit Search Run Compile Debug Project Options Window Help
SAAD13.C
temp=head;
while(temp->next!=NULL) {
temp=temp->next;
}
temp->next=newNode;
}
void mid(struct Node* head) {
int data,n,i;
struct Node *temp=head,*newNode=NULL,*ptr=NULL;
printf("\nEnter position of the Node to be inserted: ");
scanf("%i",&n);
printf("Enter data for the Node to be inserted at position %i: ",n);
scanf("%i",&data);
for(i=1;i<n-1;i++) {
temp=temp->next;
}
newNode=createnode(data);
ptr=temp->next;
temp->next=newNode;
newNode->next=ptr;
101:36
F1 Help Alt-F8 Next Msg Alt-F7 Prev Msg Alt-F9 Compile F9 Make F10 Menu
File Edit Search Run Compile Debug Project Options Window Help
SAAD13.C
}
newNode=createnode(data);
ptr=temp->next;
temp->next=newNode;
newNode->next=ptr;
}
void del(struct Node* head) {
int data;
struct Node *temp=head,*ptr=NULL,*preptr=NULL;
printf("\nEnter the data of the Node to be deleted: ");
scanf("%i",&data);
while(temp->data!=data) {
preptr=temp;
temp=temp->next;
}
ptr=temp->next;
preptr->next=ptr;
free(temp);
}
117:36
F1 Help Alt-F8 Next Msg Alt-F7 Prev Msg Alt-F9 Compile F9 Make F10 Menu

OUTPUT: -


Enter the No. of Node: 3
Enter data for Node 1: 12
Enter data for Node 2: 23
Enter data for Node 3: 34
12 -> 23 -> 34 -> NULL
Enter data for Node to be inserted at the end: 45
12 -> 23 -> 34 -> 45 -> NULL
Enter position of the Node to be inserted: 1
Enter data for the Node to be inserted at position 1: 56
12 -> 56 -> 23 -> 34 -> 45 -> NULL
Enter the data of the Node to be deleted: 56
12 -> 23 -> 34 -> 45 -> NULL

Practical Related Questions:

1. **Write a function to insert a node at the end in a Singly Linked List.**
   Ans:

```
void end(struct Node* head)
{
int data;
struct Node *temp = NULL, *newNode = NULL;
printf("\nEnter data for Node to be inserted at the end: ");
scanf("%i",&data);
newNode = createnode(data);
temp = head;
while(temp->next==NULL)
{
temp=temp->next;
}
temp->next=newNode;
}
```

2. **Write a function to check whether a singly linked list is a palindrome or not.**
   ANS:

```
int palindrome(struct node *head,int n)
{
int arr[50],i=0,j;
struct node *p=head;
while(p!=NULL)
{
arr[i++]=p->data;
p=p->next;
}
for(j=0;j<n/2;j++)
{
if(arr[j]!=arr[n-j-1])
return 0;
}
return 1;
}
```

Marks Obtained			Dated signature of Teacher
Process Related (35)	**Product Related(15)**	**Total (50)**	