DEPARTMENT OF COMPUTER ENGINEERING

Subject: - DSU	**Subject Code:** 313301
Semester: - III	**Course:** DATA STRUCTURES
Laboratory No: L003	**Name of Subject Teacher:** Prof. Imraan S.
Name of Student: Saad Sharif Kazi	**Roll Id: -** 24203A0013

Experiment No:	14
Title of Experiment	*Write a C Program to Create Two Polynomials using a Linked List.

Aim: *Write a C Program to Create Two Polynomials using a Linked List.

Algorithm:

```
Step 1: Start
Step 2: Define a structure Node having three fields: coefficient (coeff), power (power), and pointer to the next node (next).
Step 3: Declare the functions:
  createlinkedlist(int n) to create a polynomial linked list with n terms
  createnode(int coeff, int power) to allocate memory for a new node
  printList(struct Node* head) to display the polynomial
Step 4: In the main function, declare variables n1 and n2 for number of terms of two polynomials.
Step 5: Input the number of terms for the first polynomial (n1).
Step 6: Input the number of terms for the second polynomial (n2).
Step 7: Call createlinkedlist(n1) to create the first polynomial and store its head pointer in poly1.
Step 8: Call createlinkedlist(n2) to create the second polynomial and store its head pointer in poly2.
Step 9: Display the first polynomial using printList(poly1).
Step 10: Display the second polynomial using printList(poly2).
Step 11: End
```

ub-Algorithm for createlinkedlist(int n)

```
Step 1: Initialize head and temp as NULL.
Step 2: If n <= 0 then print message "Number of nodes should be greater than zero" and return NULL.
Step 3: Input coefficient and power for the first term.
Step 4: Create a new node using createnode(coeff, power) and assign it to head and temp.
Step 5: Repeat for i = 2 to n:
      a) Input coefficient and power of the ith term
      b) Create a new node using createnode(coeff, power)
      c) Link it to the list using temp->next = newNode
      d) Move temp to temp->next
```

```
Step 6: Return head
```

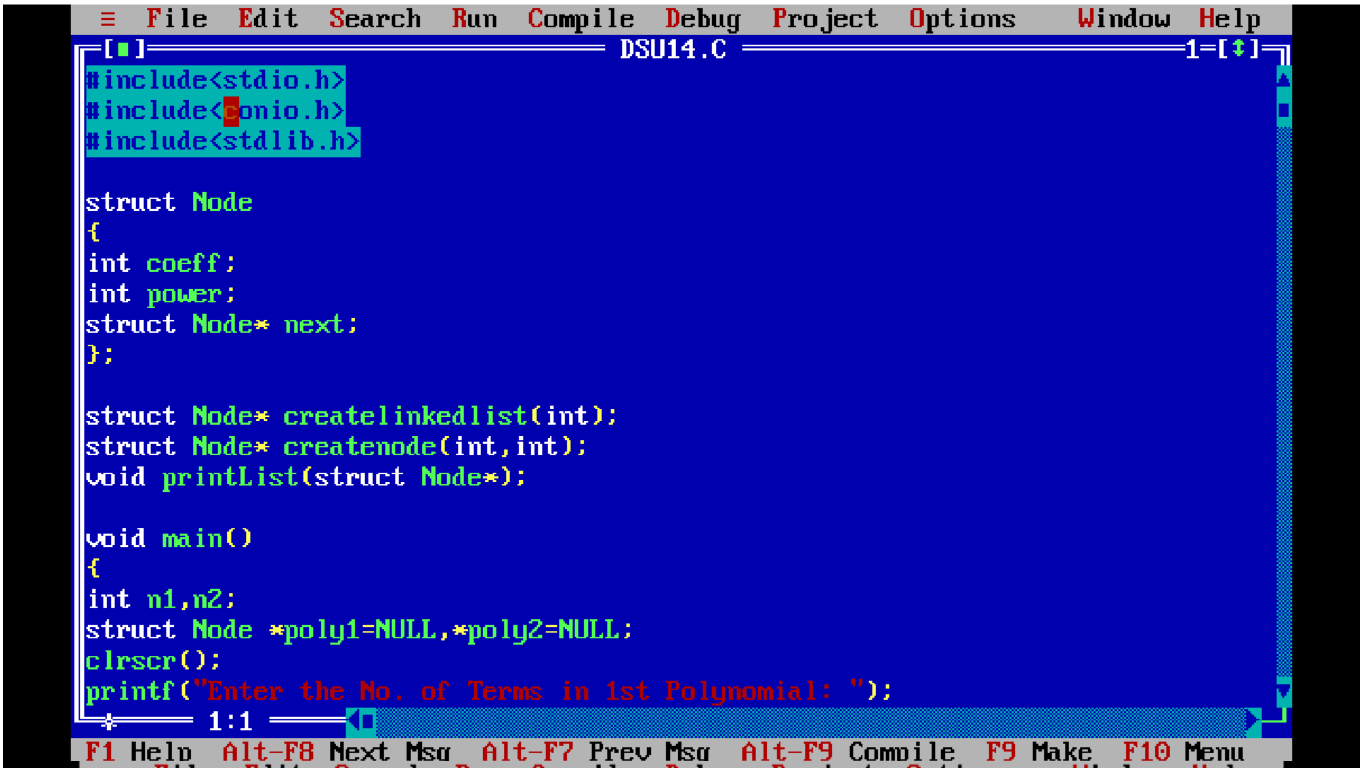
Sub-Algorithm for createnode(int coeff, int power)

```
Step 1: Allocate memory for a new node
Step 2: Assign coefficient and power to the node
Step 3: Set next = NULL
Step 4: Return new node pointer
```

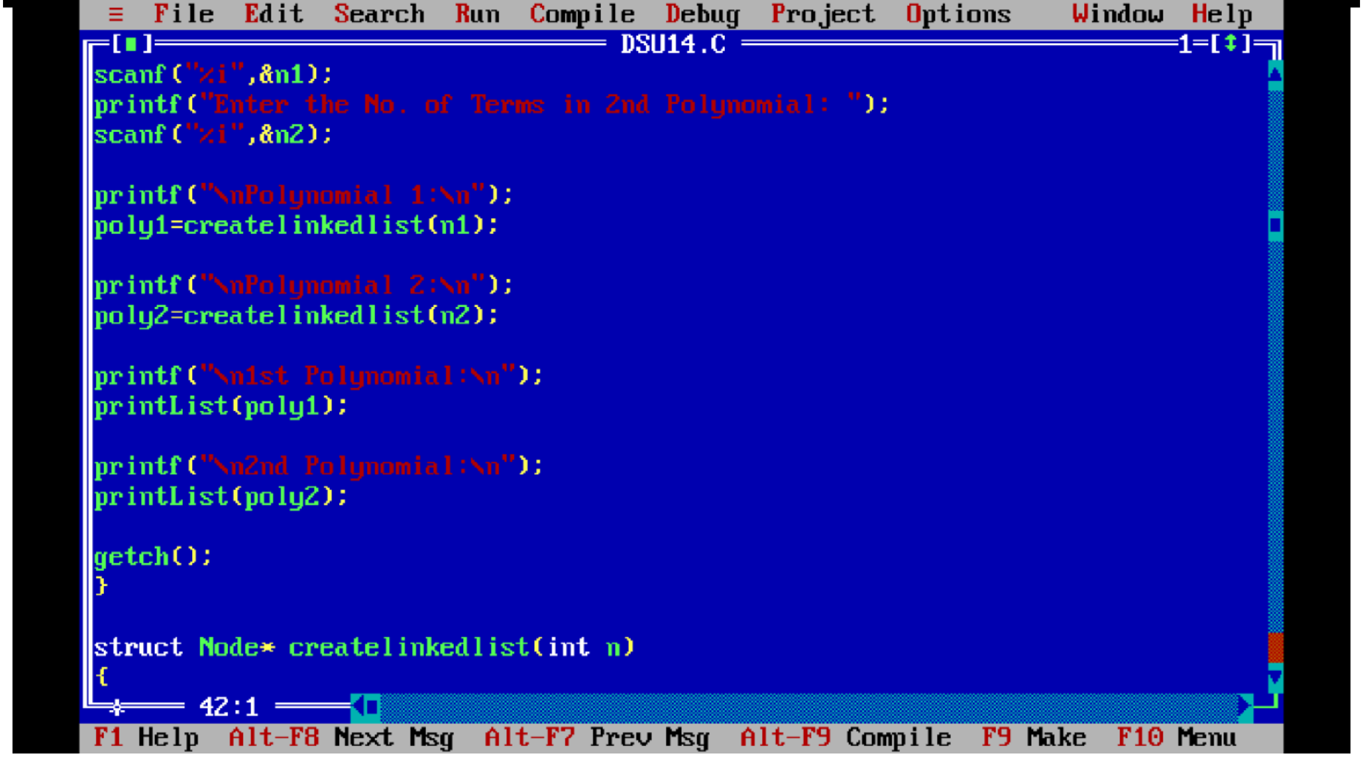
Sub-Algorithm for printList(struct Node* head)

```
Step 1: Initialize temp = head
Step 2: While temp->next != NULL, print term in the form coeff X^power + and move temp to next
Step 3: Print the last term without +
Step 4: Stop
```

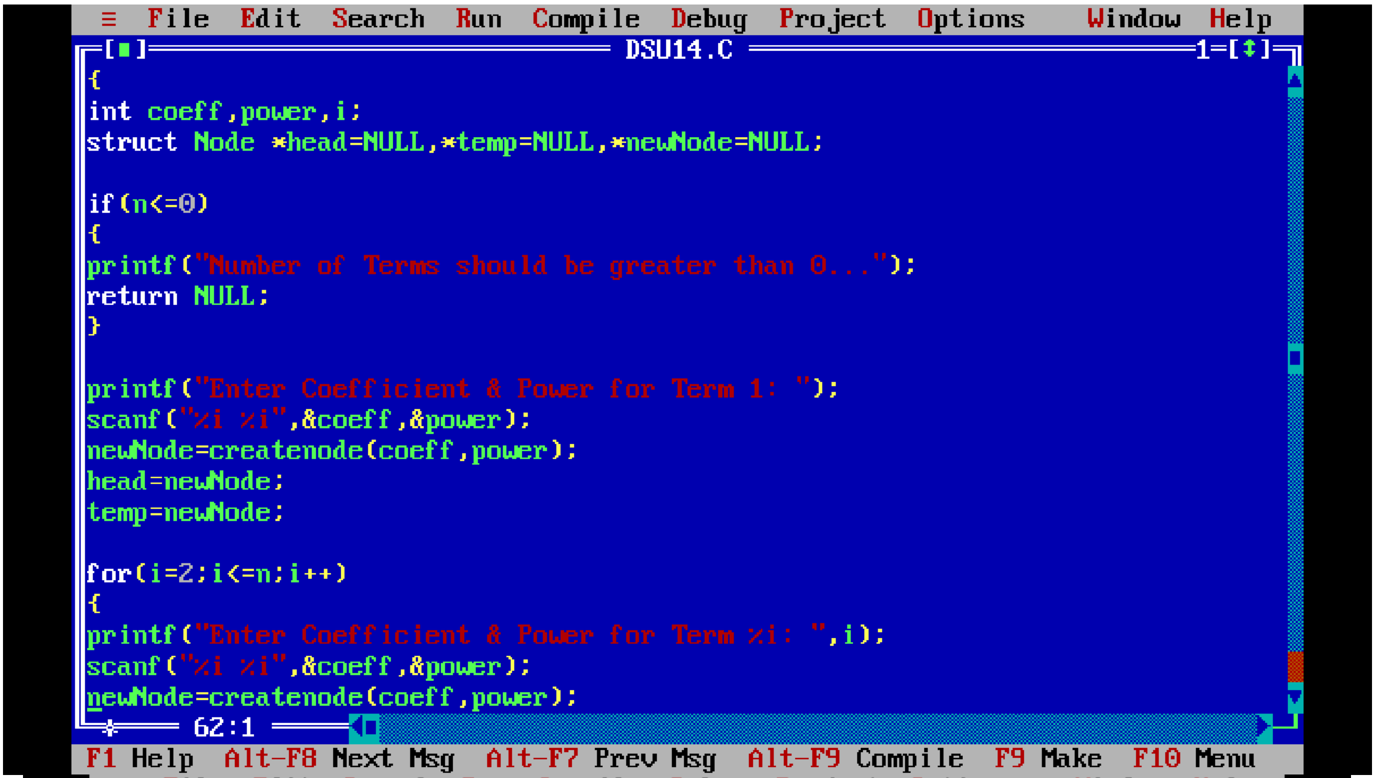
CODE:


≡ File Edit Search Run Compile Debug Project Options Window Help
[■] DSU14.C 1=[↕]
1:1
F1 Help Alt-F8 Next Msg Alt-F7 Prev Msg Alt-F9 Compile F9 Make F10 Menu

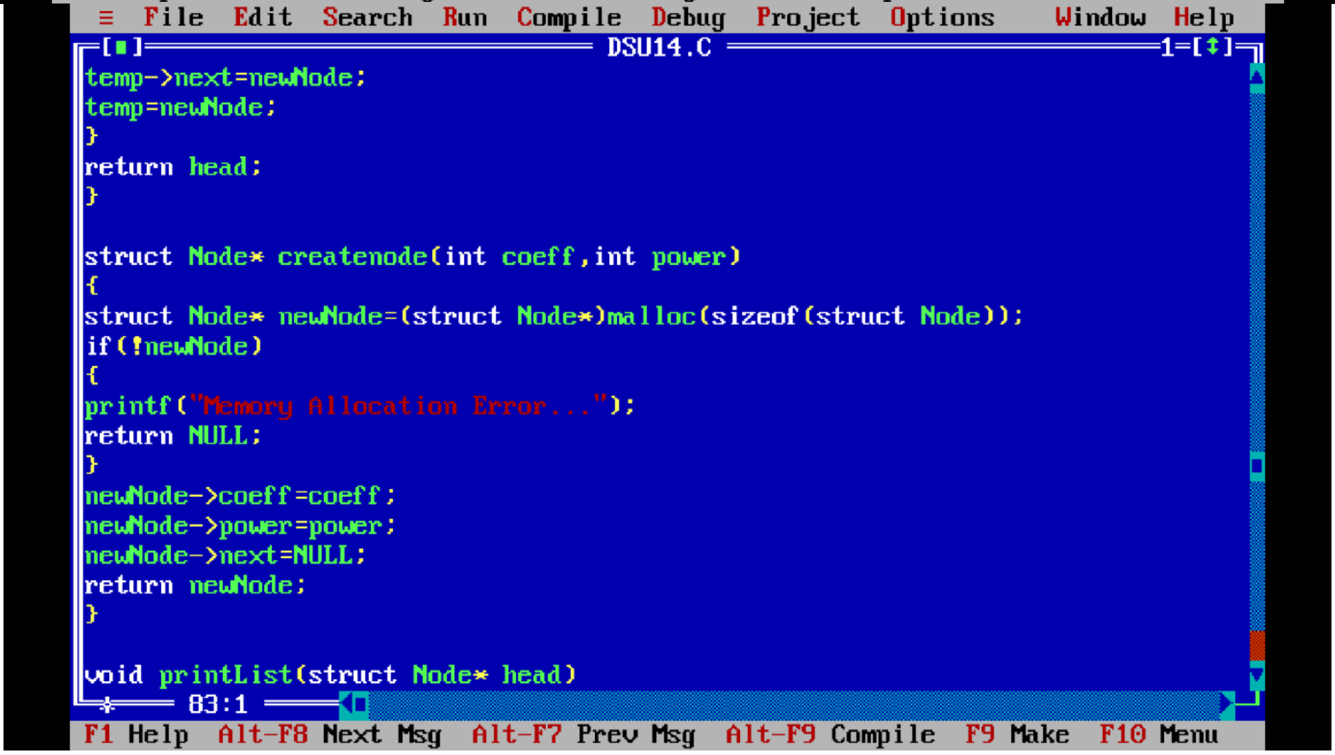
```
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>

struct Node
{
int coeff;
int power;
struct Node* next;
};

struct Node* createlinkedlist(int);
struct Node* createnode(int,int);
void printList(struct Node*);

void main()
{
int n1,n2;
struct Node *poly1=NULL,*poly2=NULL;
clrscr();
printf("Enter the No. of Terms in 1st Polynomial: ");
```


≡ File Edit Search Run Compile Debug Project Options Window Help
[■] DSU14.C 1=[↕]
42:1
F1 Help Alt-F8 Next Msg Alt-F7 Prev Msg Alt-F9 Compile F9 Make F10 Menu

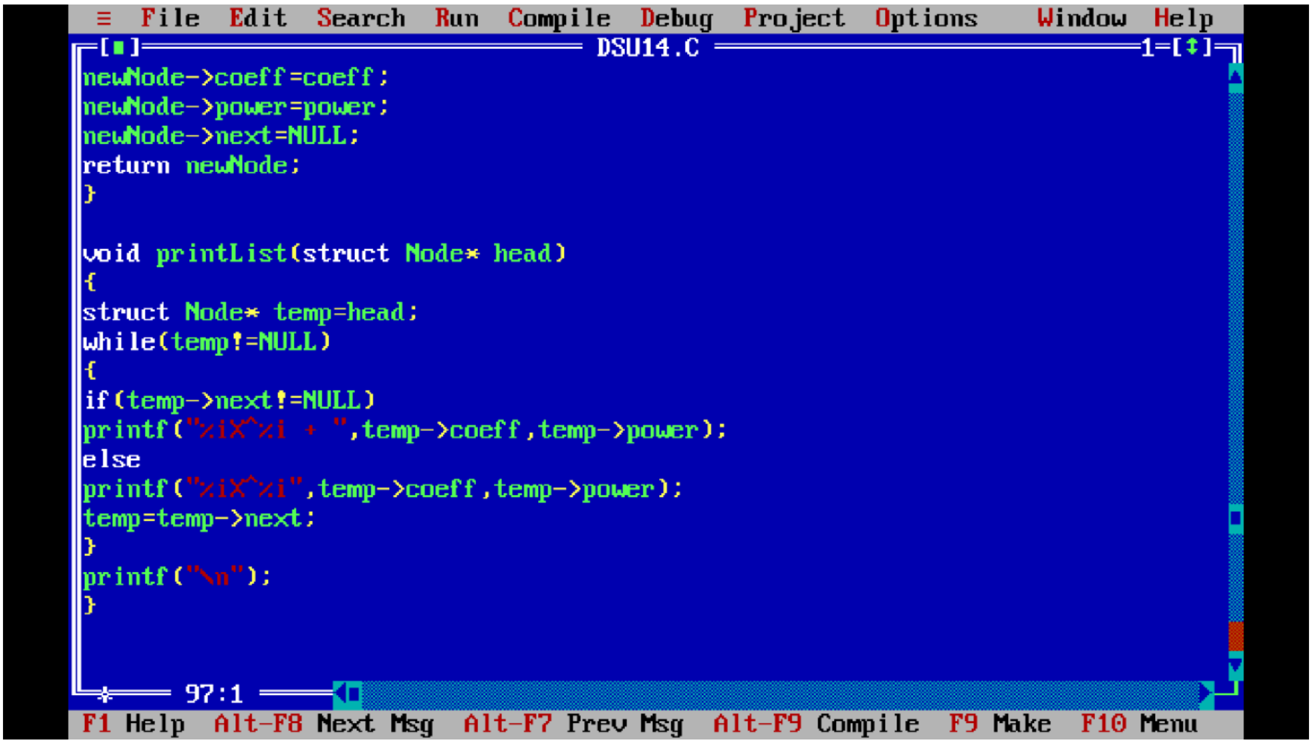
```
scanf("%i",&n1);
printf("Enter the No. of Terms in 2nd Polynomial: ");
scanf("%i",&n2);

printf("\nPolynomial 1:\n");
poly1=createlinkedlist(n1);

printf("\nPolynomial 2:\n");
poly2=createlinkedlist(n2);

printf("\n1st Polynomial:\n");
printList(poly1);

printf("\n2nd Polynomial:\n");
printList(poly2);

getch();
}

struct Node* createlinkedlist(int n)
{
```


≡ File Edit Search Run Compile Debug Project Options Window Help
DSU14.C
62:1
F1 Help Alt-F8 Next Msg Alt-F7 Prev Msg Alt-F9 Compile F9 Make F10 Menu

```
{
int coeff,power,i;
struct Node *head=NULL,*temp=NULL,*newNode=NULL;

if(n<=0)
{
printf("Number of Terms should be greater than 0...");
return NULL;
}

printf("Enter Coefficient & Power for Term 1: ");
scanf("%i %i",&coeff,&power);
newNode=createnode(coeff,power);
head=newNode;
temp=newNode;

for(i=2;i<=n;i++)
{
printf("Enter Coefficient & Power for Term %i: ",i);
scanf("%i %i",&coeff,&power);
newNode=createnode(coeff,power);
```


≡ File Edit Search Run Compile Debug Project Options Window Help
DSU14.C
83:1
F1 Help Alt-F8 Next Msg Alt-F7 Prev Msg Alt-F9 Compile F9 Make F10 Menu

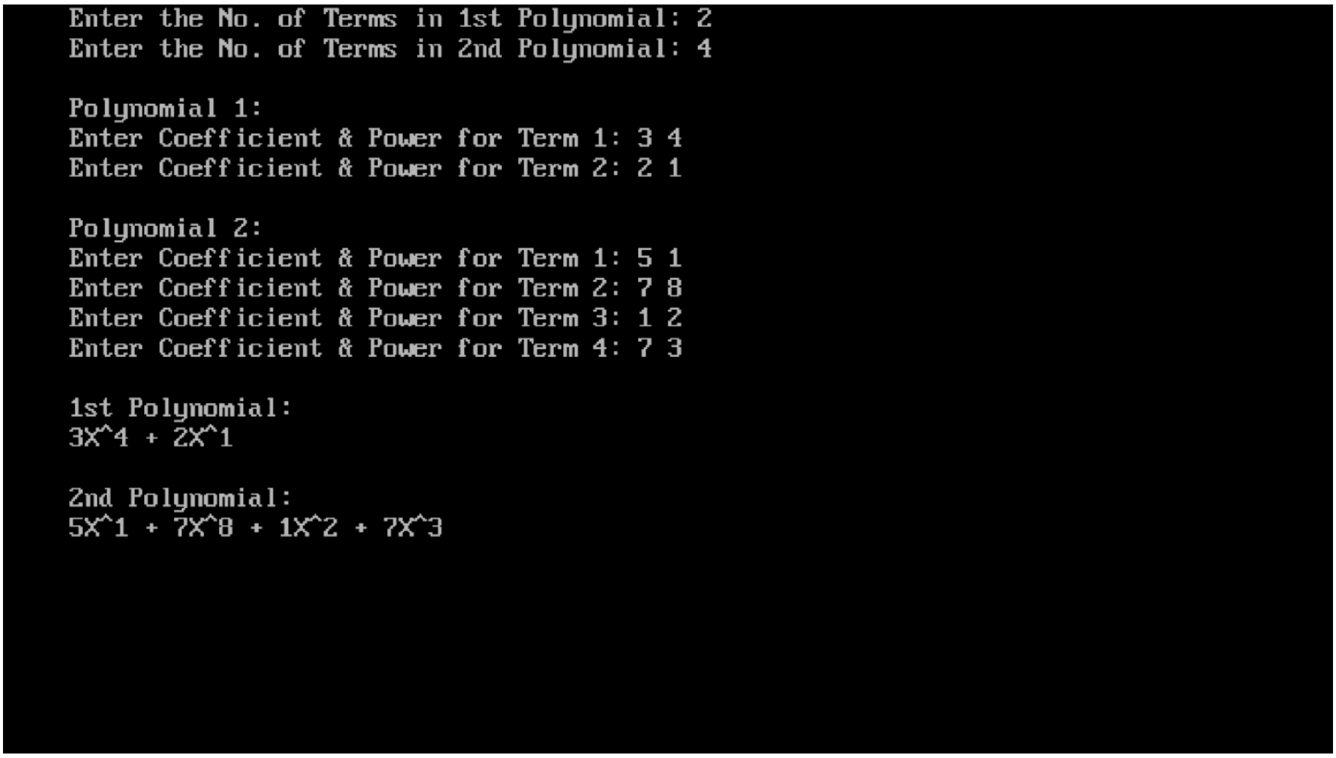
```
temp->next=newNode;
temp=newNode;
}
return head;
}

struct Node* createnode(int coeff,int power)
{
struct Node* newNode=(struct Node*)malloc(sizeof(struct Node));
if(!newNode)
{
printf("Memory Allocation Error...");
return NULL;
}
newNode->coeff=coeff;
newNode->power=power;
newNode->next=NULL;
return newNode;
}

void printList(struct Node* head)
```



≡ File Edit Search Run Compile Debug Project Options Window Help
DSU14.C
1
97:1
F1 Help Alt-F8 Next Msg Alt-F7 Prev Msg Alt-F9 Compile F9 Make F10 Menu

```
newNode->coeff=coeff;
newNode->power=power;
newNode->next=NULL;
return newNode;
}

void printList(struct Node* head)
{
struct Node* temp=head;
while(temp!=NULL)
{
if(temp->next!=NULL)
printf("%iX^%i + ",temp->coeff,temp->power);
else
printf("%iX^%i",temp->coeff,temp->power);
temp=temp->next;
}
printf("\n");
}
```

OUTPUT: -



```
Enter the No. of Terms in 1st Polynomial: 2
Enter the No. of Terms in 2nd Polynomial: 4

Polynomial 1:
Enter Coefficient & Power for Term 1: 3 4
Enter Coefficient & Power for Term 2: 2 1

Polynomial 2:
Enter Coefficient & Power for Term 1: 5 1
Enter Coefficient & Power for Term 2: 7 8
Enter Coefficient & Power for Term 3: 1 2
Enter Coefficient & Power for Term 4: 7 3

1st Polynomial:
3X^4 + 2X^1

2nd Polynomial:
5X^1 + 7X^8 + 1X^2 + 7X^3
```

<u>Practical Related Questions</u>:

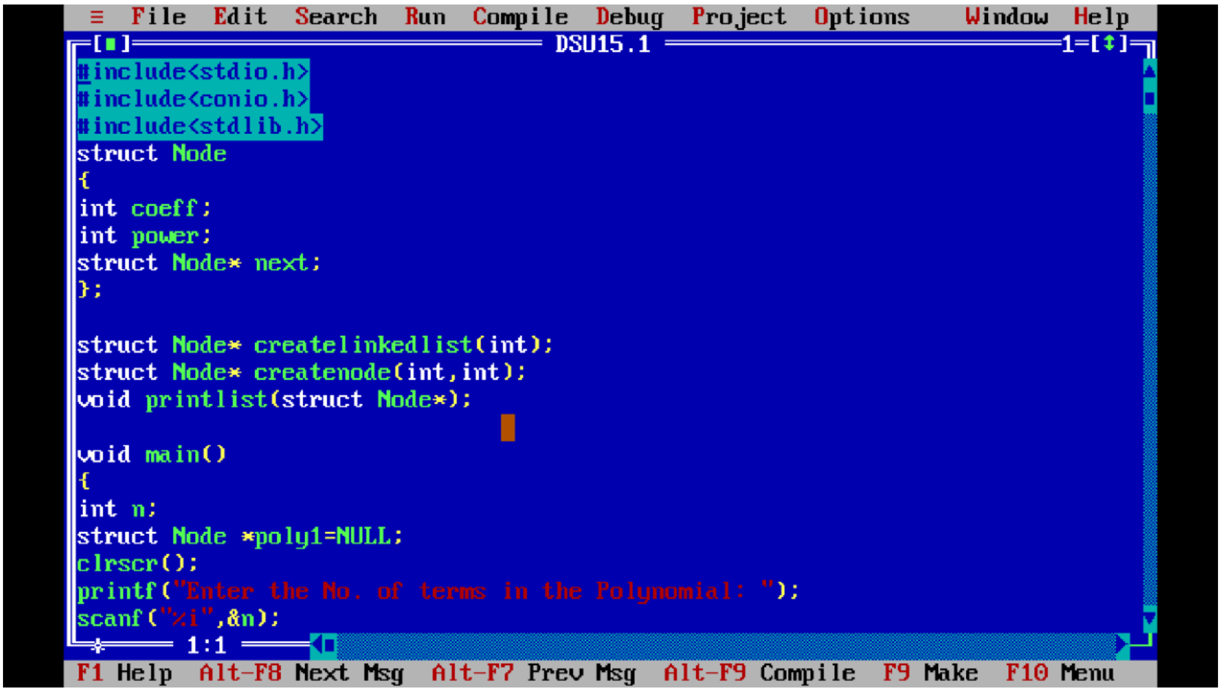
1. Write a node structure to represent a polynomial using linked list

Ans:

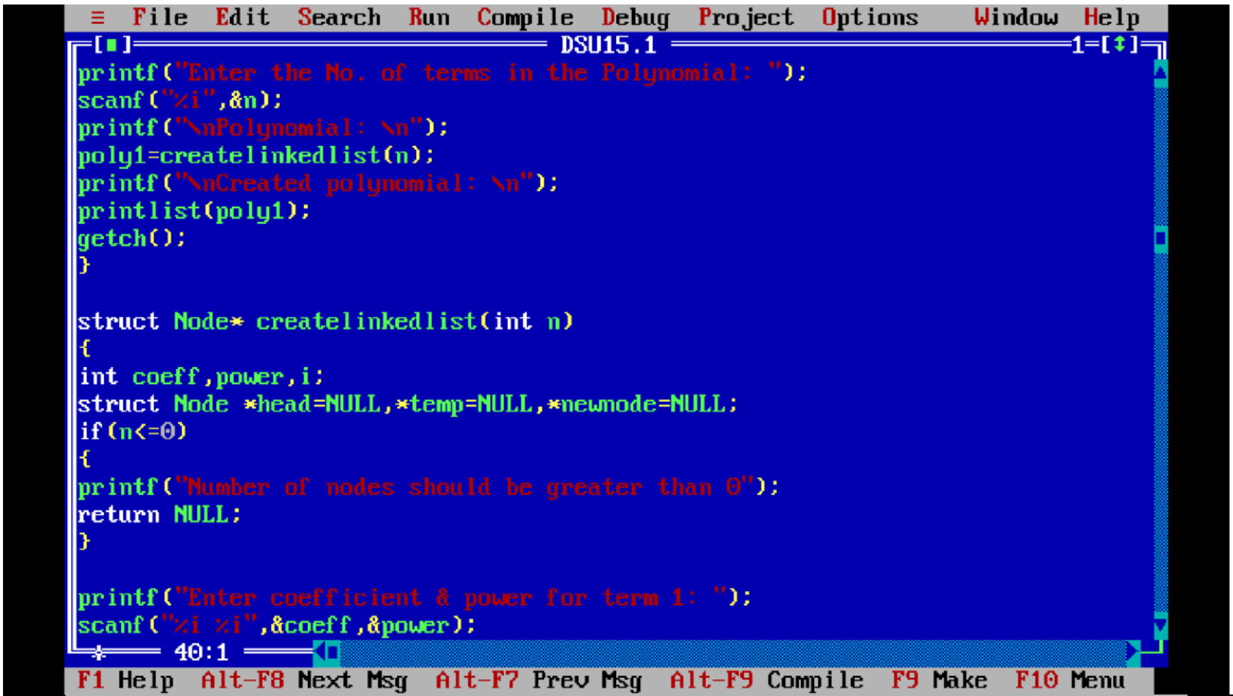
```
struct Node
{
int coefficient;
int power;
struct Node* next
};
```

2. Write a C program to Create a polynomial 5x4 + 3x2 + 1.

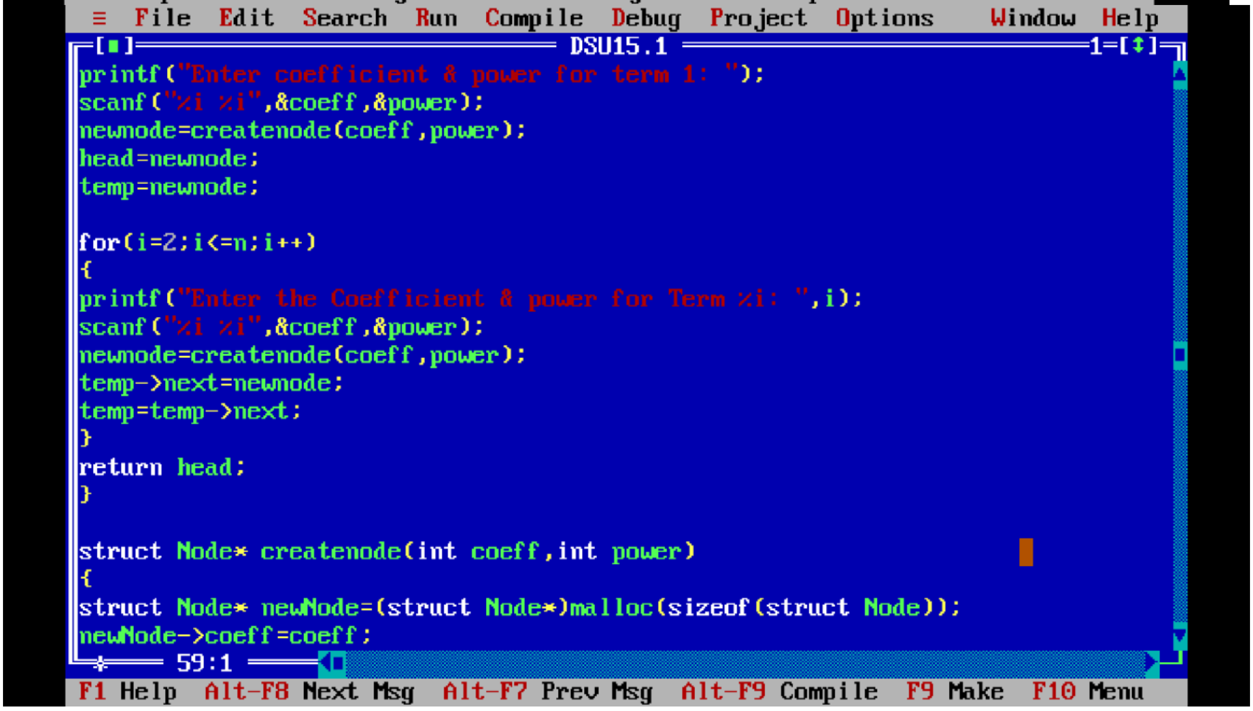
Ans:


≡ File Edit Search Run Compile Debug Project Options Window Help
[■] DSU15.1 1=[↕]
1:1
F1 Help Alt-F8 Next Msg Alt-F7 Prev Msg Alt-F9 Compile F9 Make F10 Menu

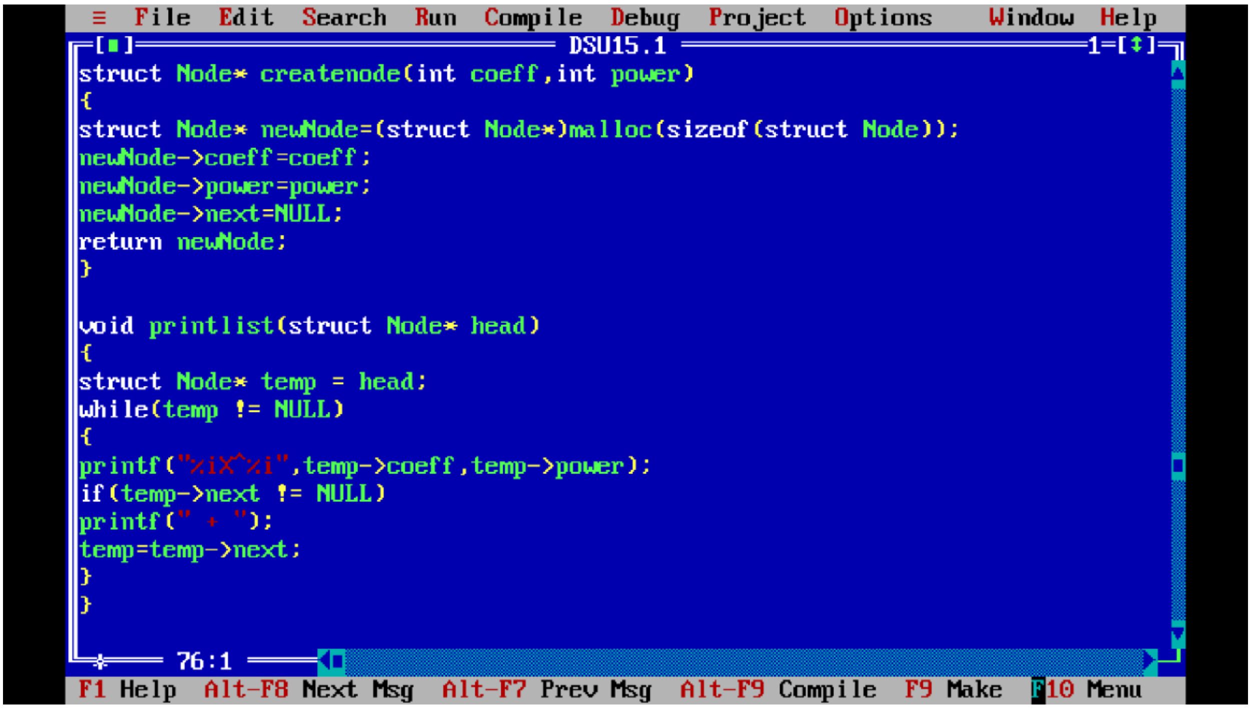
```
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
struct Node
{
int coeff;
int power;
struct Node* next;
};

struct Node* createlinkedlist(int);
struct Node* createnode(int,int);
void printlist(struct Node*);

void main()
{
int n;
struct Node *poly1=NULL;
clrscr();
printf("Enter the No. of terms in the Polynomial: ");
scanf("%i",&n);
```


≡ File Edit Search Run Compile Debug Project Options Window Help
DSU15.1
1=[↕]
40:1
F1 Help Alt-F8 Next Msg Alt-F7 Prev Msg Alt-F9 Compile F9 Make F10 Menu

```
printf("Enter the No. of terms in the Polynomial: ");
scanf("%i",&n);
printf("\nPolynomial: \n");
poly1=createlinkedlist(n);
printf("\nCreated polynomial: \n");
printlist(poly1);
getch();
}

struct Node* createlinkedlist(int n)
{
int coeff,power,i;
struct Node *head=NULL,*temp=NULL,*newnode=NULL;
if(n<=0)
{
printf("Number of nodes should be greater than 0");
return NULL;
}

printf("Enter coefficient & power for term 1: ");
scanf("%i %i",&coeff,&power);
```

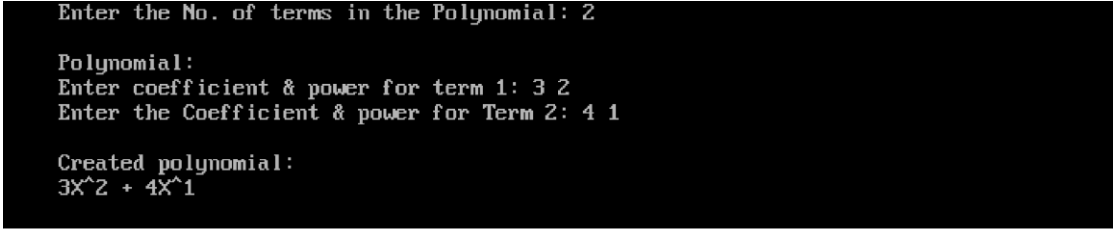

≡ File Edit Search Run Compile Debug Project Options Window Help
DSU15.1
1=[↕]
59:1
F1 Help Alt-F8 Next Msg Alt-F7 Prev Msg Alt-F9 Compile F9 Make F10 Menu

```
printf("Enter coefficient & power for term 1: ");
scanf("%i %i",&coeff,&power);
newnode=createnode(coeff,power);
head=newnode;
temp=newnode;

for(i=2;i<=n;i++)
{
printf("Enter the Coefficient & power for Term %i: ",i);
scanf("%i %i",&coeff,&power);
newnode=createnode(coeff,power);
temp->next=newnode;
temp=temp->next;
}
return head;
}

struct Node* createnode(int coeff,int power)
{
struct Node* newNode=(struct Node*)malloc(sizeof(struct Node));
newNode->coeff=coeff;
```


≡ File Edit Search Run Compile Debug Project Options Window Help
[■] DSU15.1 1=[↕]
76:1
F1 Help Alt-F8 Next Msg Alt-F7 Prev Msg Alt-F9 Compile F9 Make F10 Menu

```c
struct Node* createnode(int coeff,int power)
{
struct Node* newNode=(struct Node*)malloc(sizeof(struct Node));
newNode->coeff=coeff;
newNode->power=power;
newNode->next=NULL;
return newNode;
}

void printlist(struct Node* head)
{
struct Node* temp = head;
while(temp != NULL)
{
printf("%iX^%i",temp->coeff,temp->power);
if(temp->next != NULL)
printf(" + ");
temp=temp->next;
}
}
```

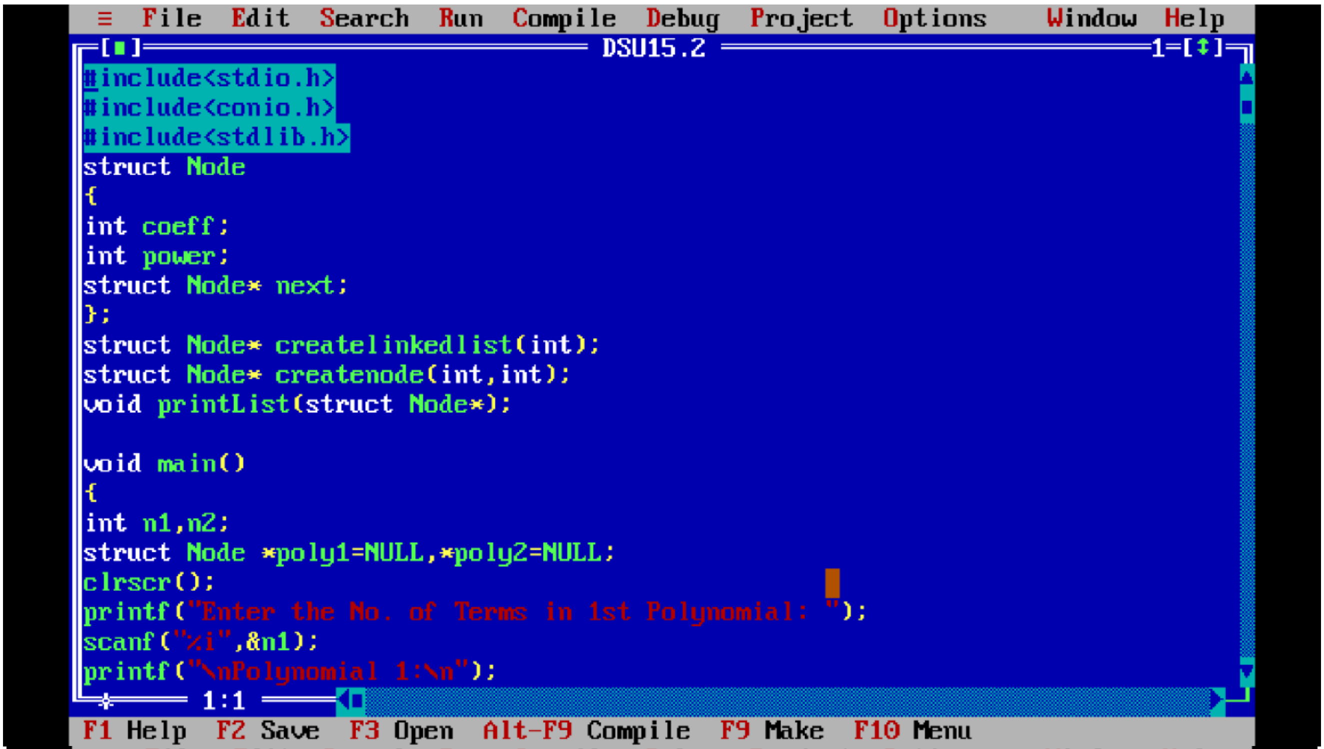
Output:



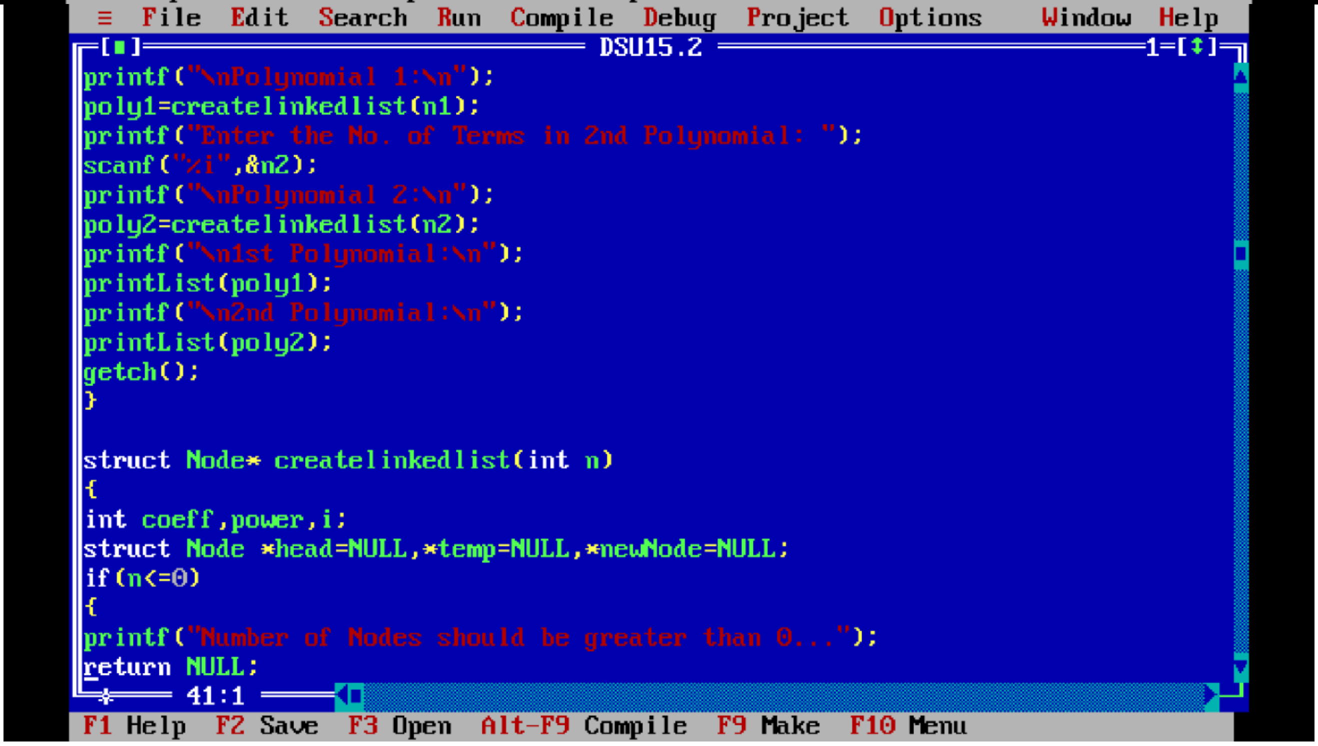
```
Enter the No. of terms in the Polynomial: 2

Polynomial:
Enter coefficient & power for term 1: 3 2
Enter the Coefficient & power for Term 2: 4 1

Created polynomial:
3X^2 + 4X^1
```

3. Write a C program to display created polynomials.

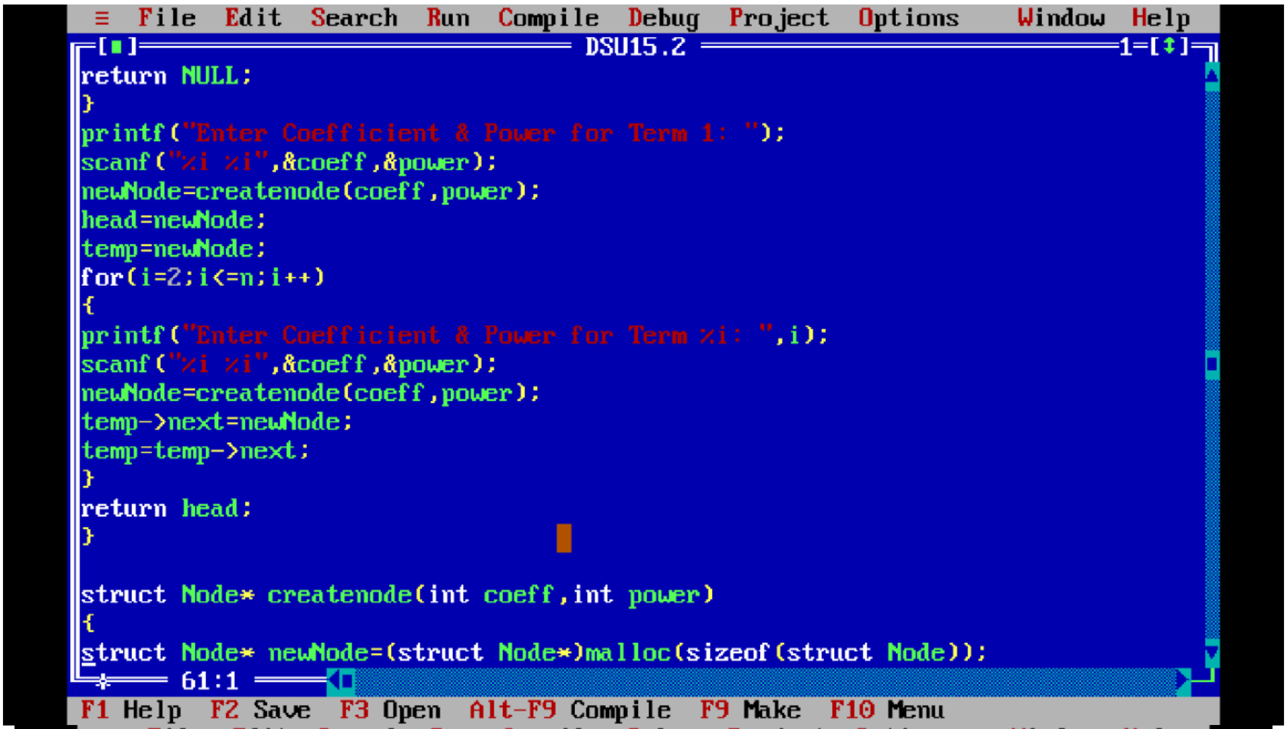
Ans:


≡ File Edit Search Run Compile Debug Project Options Window Help
[■] DSU15.2 1=[↕]
1:1
F1 Help F2 Save F3 Open Alt-F9 Compile F9 Make F10 Menu

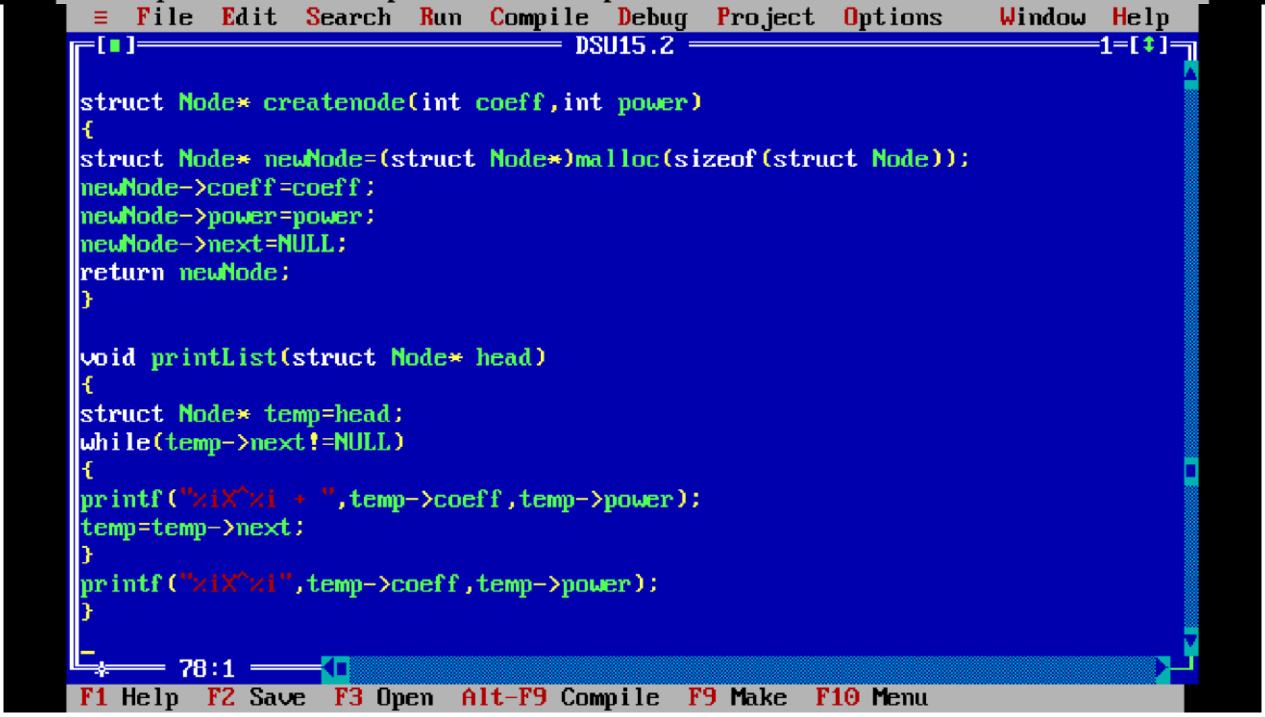
```
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
struct Node
{
int coeff;
int power;
struct Node* next;
};
struct Node* createlinkedlist(int);
struct Node* createnode(int,int);
void printList(struct Node*);

void main()
{
int n1,n2;
struct Node *poly1=NULL,*poly2=NULL;
clrscr();
printf("Enter the No. of Terms in 1st Polynomial: ");
scanf("%i",&n1);
printf("\nPolynomial 1:\n");
```


≡ File Edit Search Run Compile Debug Project Options Window Help
[■] DSU15.2 1=[↕]
41:1
F1 Help F2 Save F3 Open Alt-F9 Compile F9 Make F10 Menu

```
printf("\nPolynomial 1:\n");
poly1=createlinkedlist(n1);
printf("Enter the No. of Terms in 2nd Polynomial: ");
scanf("%i",&n2);
printf("\nPolynomial 2:\n");
poly2=createlinkedlist(n2);
printf("\n1st Polynomial:\n");
printList(poly1);
printf("\n2nd Polynomial:\n");
printList(poly2);
getch();
}

struct Node* createlinkedlist(int n)
{
int coeff,power,i;
struct Node *head=NULL,*temp=NULL,*newNode=NULL;
if(n<=0)
{
printf("Number of Nodes should be greater than 0...");
return NULL;
```

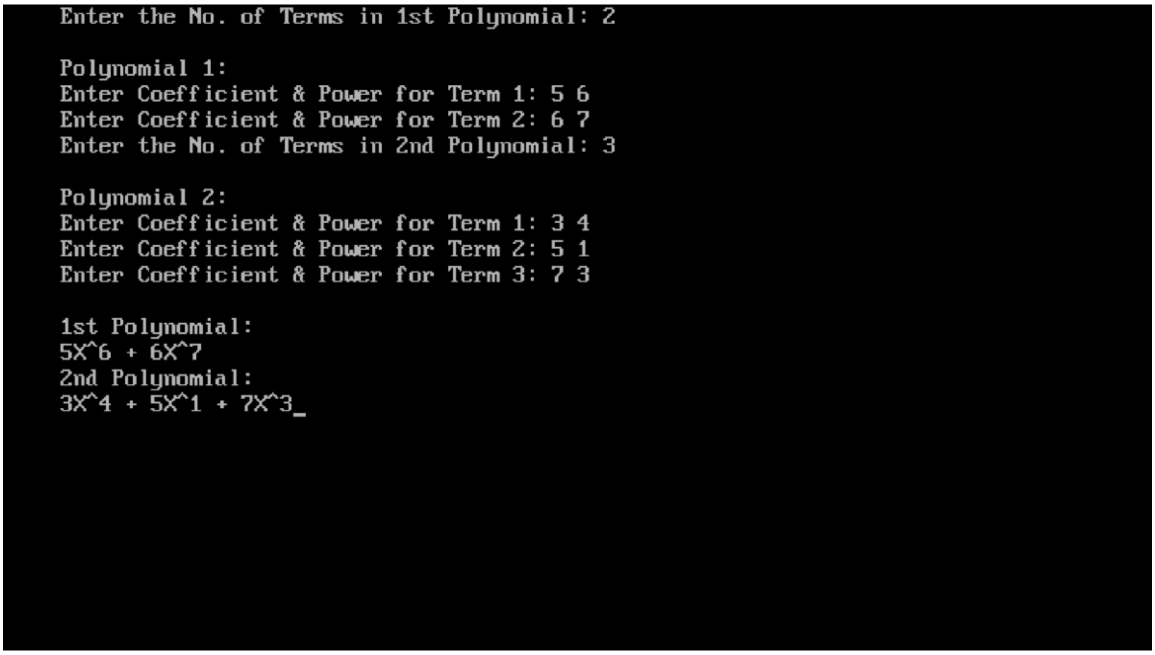

≡ File Edit Search Run Compile Debug Project Options Window Help
[■] DSU15.2 1=[↕]
61:1
F1 Help F2 Save F3 Open Alt-F9 Compile F9 Make F10 Menu

```
return NULL;
}
printf("Enter Coefficient & Power for Term 1: ");
scanf("%i %i",&coeff,&power);
newNode=createnode(coeff,power);
head=newNode;
temp=newNode;
for(i=2;i<=n;i++)
{
printf("Enter Coefficient & Power for Term %i: ",i);
scanf("%i %i",&coeff,&power);
newNode=createnode(coeff,power);
temp->next=newNode;
temp=temp->next;
}
return head;
}

struct Node* createnode(int coeff,int power)
{
struct Node* newNode=(struct Node*)malloc(sizeof(struct Node));
```


≡ File Edit Search Run Compile Debug Project Options Window Help
[■] DSU15.2 1=[↕]
78:1
F1 Help F2 Save F3 Open Alt-F9 Compile F9 Make F10 Menu

```

struct Node* createnode(int coeff,int power)
{
struct Node* newNode=(struct Node*)malloc(sizeof(struct Node));
newNode->coeff=coeff;
newNode->power=power;
newNode->next=NULL;
return newNode;
}

void printList(struct Node* head)
{
struct Node* temp=head;
while(temp->next!=NULL)
{
printf("%iX^%i + ",temp->coeff,temp->power);
temp=temp->next;
}
printf("%iX^%i",temp->coeff,temp->power);
}
```

OUTPUT: -



```
Enter the No. of Terms in 1st Polynomial: 2

Polynomial 1:
Enter Coefficient & Power for Term 1: 5 6
Enter Coefficient & Power for Term 2: 6 7
Enter the No. of Terms in 2nd Polynomial: 3

Polynomial 2:
Enter Coefficient & Power for Term 1: 3 4
Enter Coefficient & Power for Term 2: 5 1
Enter Coefficient & Power for Term 3: 7 3

1st Polynomial:
5X^6 + 6X^7
2nd Polynomial:
3X^4 + 5X^1 + 7X^3_
```

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