Name	Komal Tarachandani
UID no.	2021600065
Experiment No.	3

AIM:	Implement polynomial expression through linked lists
Program3	
PROBLEM STATEMENT :	Implement Polynomial Expression execution using Singly Linked List P1(X)=3*X^4+12* x^2+10 P2(x)=5X^3-4*x^2+3 1- Create Linked Representation of Polynomial Expressions. Display the same 2-Perform arithmetic operations on the given expressions. Shoe the linked representation of the resulting polynomial expression.

PROGRAM:	#include <iostream>
	
	using namespace std;
	class polynode
	{
	public:
	int expo;
	int coeff;
	polynode *next;
	};
	polynode *cur,*head;
	void create_node(int coefficient,int exponent)
	{
	polynode *newnode;
	newnode=new polynode();
	newnode->expo=exponent;
	newnode->coeff=coefficient;
	newnode->next=NULL;
	if(head==NULL)
	{

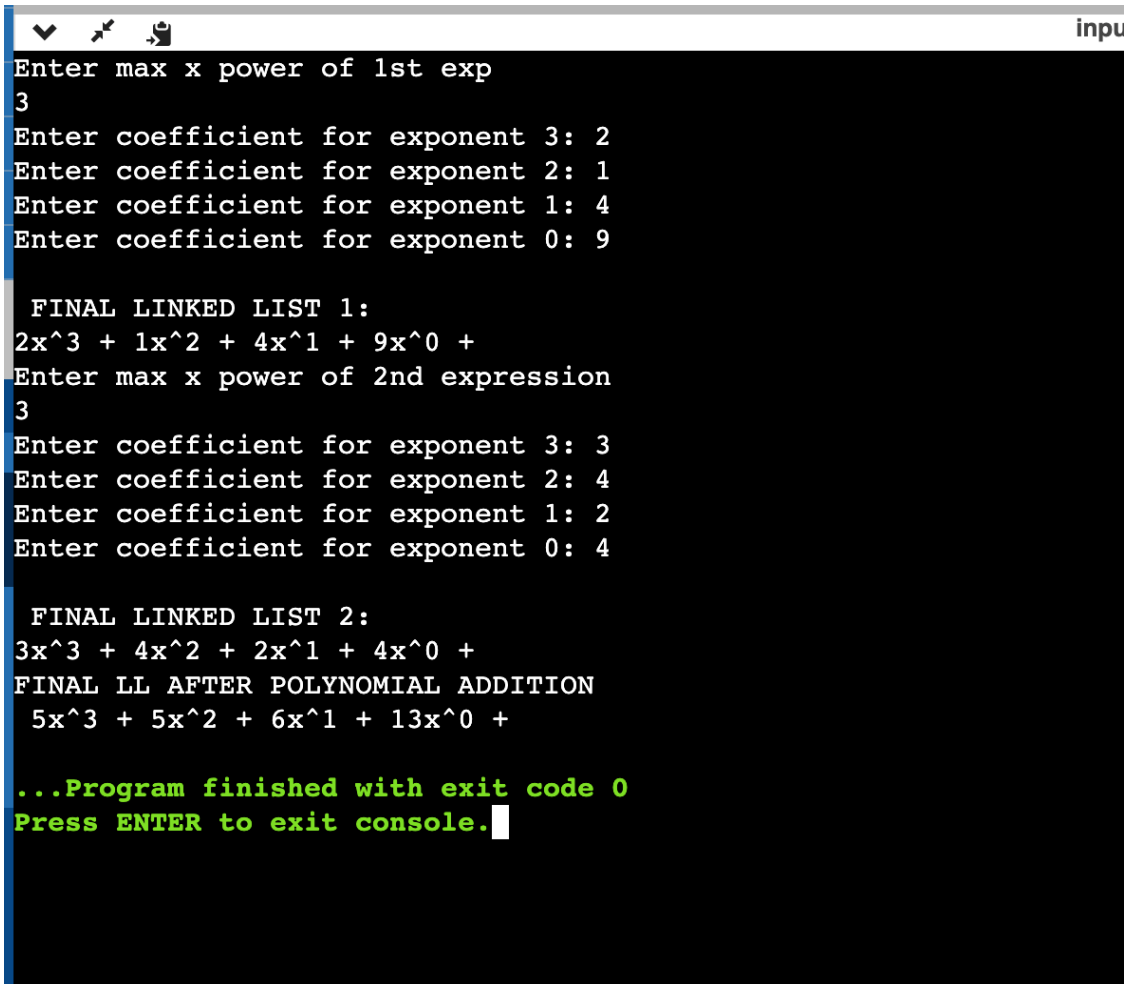
```
            head=newnode;
            cur=newnode;
        }
        else
        {
            cur->next=newnode;
            cur=cur->next;
        }
}
void display(polynode *ptr)
{
    while(ptr!=NULL)
    {
        cout<<ptr->coeff<<"x^"<<ptr->expo<<" + ";
        ptr=ptr->next;
    }
}
int main()
{
    polynode *newnode,*newnode2,*head1,*cur1,*cur2,*ptr1;
    head=NULL;
    int n1,n2,coefficient;
    cout<<"Enter max x power of 1st exp"<<endl;
    cin>>n1;
    for(int i=n1;i>=0;i--)
    {
        cout<<"Enter coefficient for exponent "<<i<<": ";
        cin>>coefficient;
        create_node(coefficient,i);
    }
    cur=head;
    newnode2=new polynode();
    newnode2->next=NULL;
    newnode2->expo=cur->expo;
    newnode2->coeff=cur->coeff;
    head1=newnode2;
    ptr1=newnode2;
    while(cur->next!=NULL)
    {
        cur=cur->next;
```

```
    newnode2=new polynode();
    newnode2->next=NULL;
    newnode2->expo=cur->expo;
    newnode2->coeff=cur->coeff;
    ptr1->next=newnode2;
    ptr1=ptr1->next;
  }
  cout<<"\n FINAL LINKED LIST 1:\n";
  head=NULL;
  display(head1);
  cout<<"\nEnter max x power of 2nd expression "<<endl;
  cin>>n2;
  for(int i=n2;i>=0;i--)
  {
    cout<<"Enter coefficient for exponent "<<i<<": ";
    cin>>coefficient;
    create_node(coefficient,i);
  }
  cout<<"\n FINAL LINKED LIST 2:\n";
  display(head);

  int n=min(n1,n2);
  cur1=head1;
  cur2=head;
  while(cur1!=NULL || cur2!=NULL)
  {
    if(cur1->expo==cur2->expo)
    {
      int temp=cur1->coeff;
      cur1->coeff=cur2->coeff+cur1->coeff;
      cur2->coeff=cur2->coeff+temp;
      cur1=cur1->next;
      cur2=cur2->next;
    }
    else if(cur1->expo>cur2->expo)
    {
      cur1=cur1->next;
    }
    else
    {
```

```
            cur2=cur2->next;
        }
    }
    cout<<" \nFINAL LL AFTER POLYNOMIAL ADDITION\n ";
    if(n==n1) display(head);
    else display(head1);
    return 0;
}
```

RESULT:



CONCLUSION:	Hence I was able to learn the application of linked lists.