Assignment 4

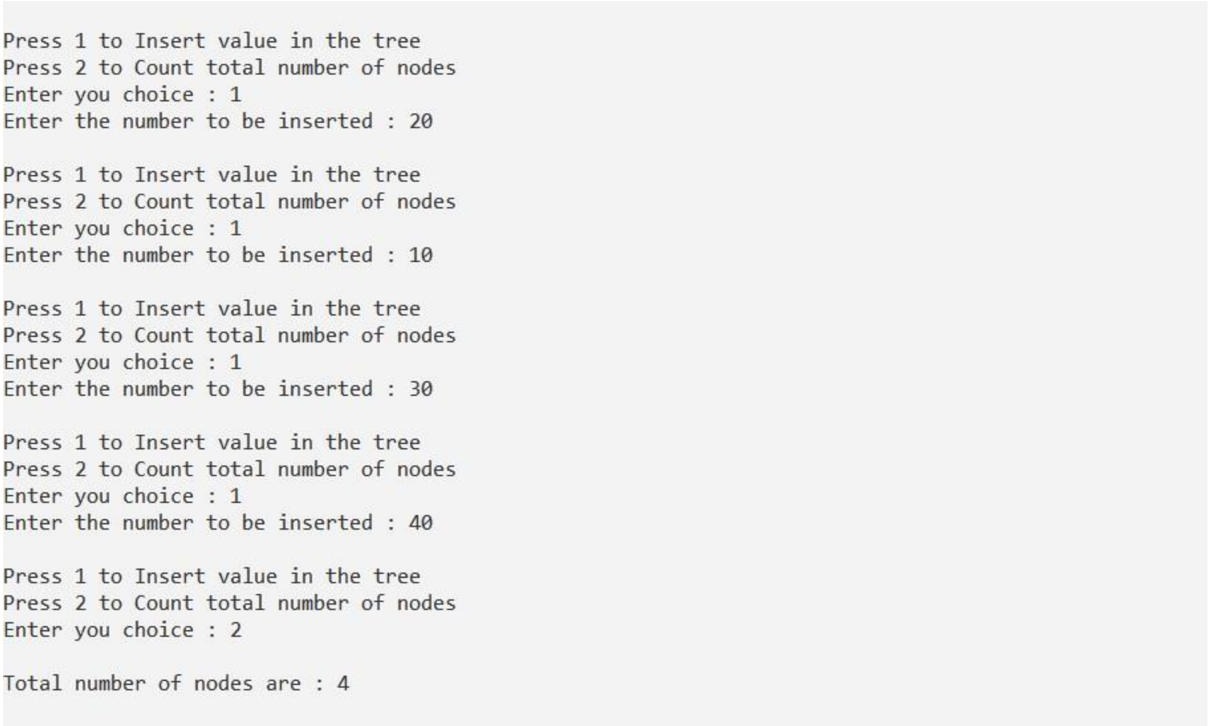
Name: Bharat Upadhyay
2017641
PCS 302(Data Structures with C)
Section : AI & DS

Q1 – Assuming that we already have a Binary Search Tree with address root. Write a function to count total number of nodes and print it in the main function.

Ans:

```
//Bharat Upadhyay
//2017641

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

typedef struct node
{
   int info;
   struct node *left,*right;
}tree;

void insert(tree**,int num);
int count(tree*);

int main()
{
   tree *root=NULL;
   int ch,num;
   do
   {
      printf("\nPress 1 to Insert value in the tree");
      printf("\nPress 2 to Count total number of nodes");
      printf("\nEnter you choice : ");
      scanf("%d",&ch);
      switch(ch)
      {
      case 1:
         printf("Enter the number to be inserted : ");
         scanf("%d",&num);
         insert(&root,num);
         break;
      case 2:
         printf("\nTotal number of nodes are : %d \n",count(root));
         break;
      default:
         printf("\nInvalid Choice");
         break;
      }
   }while(ch<=2);
}

void insert(tree** rt,int num)
{
   tree* p;
```

```
  if(*rt==NULL)
  {
    p=(tree*)malloc(sizeof(tree));
    p->info=num;
    (p)->left=(p)->right=NULL;
    *rt=p;
  }
  else
  {
    if(num<(*rt)->info)
      insert(&(*rt)->left,num);
    else
      insert(&(*rt)->right,num);
  }
}


int count(tree *rt)
{
  int c=1;
  if(rt==NULL)
    return 0;
  else
  {
    c+=count(rt->left);
    c+=count(rt->right);
    return c;
  }
}
```



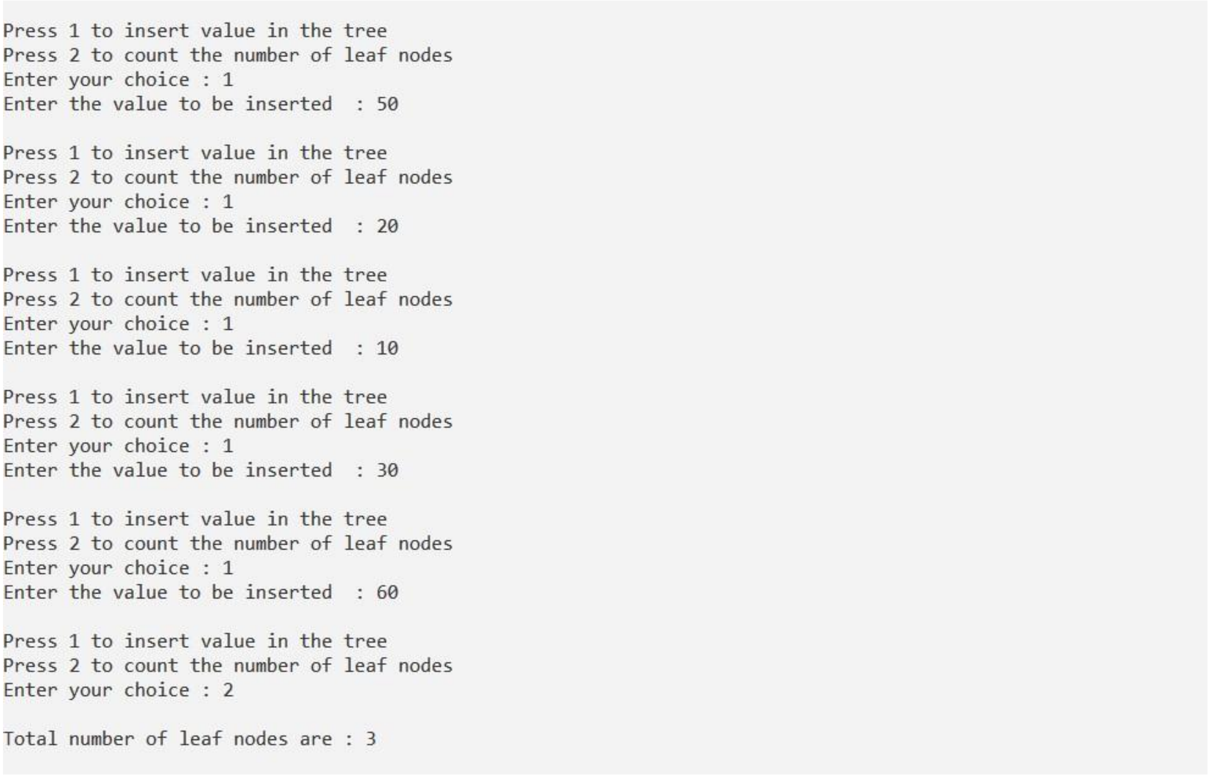
```
Press 1 to Insert value in the tree
Press 2 to Count total number of nodes
Enter you choice : 1
Enter the number to be inserted : 20

Press 1 to Insert value in the tree
Press 2 to Count total number of nodes
Enter you choice : 1
Enter the number to be inserted : 10

Press 1 to Insert value in the tree
Press 2 to Count total number of nodes
Enter you choice : 1
Enter the number to be inserted : 30

Press 1 to Insert value in the tree
Press 2 to Count total number of nodes
Enter you choice : 1
Enter the number to be inserted : 40

Press 1 to Insert value in the tree
Press 2 to Count total number of nodes
Enter you choice : 2

Total number of nodes are : 4
```

Q2 – Assuming that we have a Binary Search Tree. Write a function to count all leaf nodes.

Ans:

```
//Bharat Upadhyay
//2017641

#include <stdio.h>
#include <stdlib.h>
typedef struct node
{
   int info;
   struct node *left,*right;
}tree;

void insert(tree**,int);
int count_leaf(tree*);

int main()
{
   tree *root=NULL;
   int ch,num;
   do
   {
      printf("\nPress 1 to insert value in the tree");
      printf("\nPress 2 to count the number of leaf nodes ");
      printf("\nEnter your choice : ");
      scanf("%d",&ch);
      switch(ch)
      {
      case 1:
         printf("Enter the value to be inserted  : ");
         scanf("%d",&num);
         insert(&root,num);
         break;
      case 2:
         printf("\nTotal number of leaf nodes are : %d \n",count_leaf(root));
         break;
      default:
         printf("\nInvalid choice");
         break;
      }
   }while(ch<=2);
}

void insert(tree **rt,int num)
{
   tree *p=NULL;
   if(*rt==NULL)
   {
      p=(tree*)malloc(sizeof(tree));
      p->info=num;
      p->left=p->right=NULL;
      *rt=p;
   }
   else
   {
      if(num<(*rt)->info)
         insert(&(*rt)->left,num);
```

```
        else
            insert(&(*rt)->right,num);
    }
}


int count_leaf(tree* rt)
{
    if(rt==NULL)
    {
        return 0;
    }
    if(rt->right==NULL && rt->left==NULL)
    {
        return 1;
    }
    else
    {
        return count_leaf(rt->left)+count_leaf(rt->right);
    }
}
```

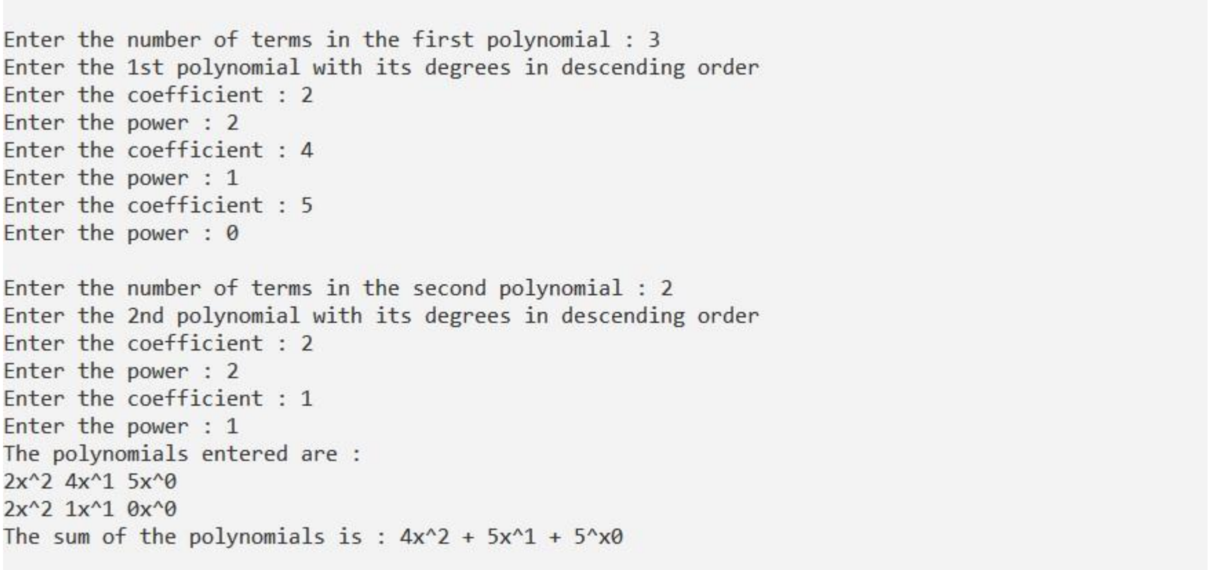


```
Press 1 to insert value in the tree
Press 2 to count the number of leaf nodes
Enter your choice : 1
Enter the value to be inserted  : 50

Press 1 to insert value in the tree
Press 2 to count the number of leaf nodes
Enter your choice : 1
Enter the value to be inserted  : 20

Press 1 to insert value in the tree
Press 2 to count the number of leaf nodes
Enter your choice : 1
Enter the value to be inserted  : 10

Press 1 to insert value in the tree
Press 2 to count the number of leaf nodes
Enter your choice : 1
Enter the value to be inserted  : 30

Press 1 to insert value in the tree
Press 2 to count the number of leaf nodes
Enter your choice : 1
Enter the value to be inserted  : 60

Press 1 to insert value in the tree
Press 2 to count the number of leaf nodes
Enter your choice : 2

Total number of leaf nodes are : 3
```

Q3 – Using linked list, implement addition of two polynomials. Sum of polynomials should be stored using existing links.

```
//Bharat Upadhyay
//2017641

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

typedef struct node
{
  int coef,power;
  struct node *next;
}nodetype;

void insert(nodetype**,nodetype **);
void calc(nodetype*,nodetype*);
void set_value(nodetype **);
void display(nodetype *);

int main()
{
  nodetype *l1=NULL,*r1=NULL,*l2=NULL,*r2=NULL,*l3=NULL;
  int i=0,n=0;
  printf("\nEnter the number of terms in the first polynomial : ");
  scanf("%d",&n);
  printf("Enter the 1st polynomial with its degrees in descending order\n");
  for(i=0;i<n;i++)
    {
      insert(&l1,&r1);
    }
  printf("\nEnter the number of terms in the second polynomial : ");
  scanf("%d",&n);
  printf("Enter the 2nd polynomial with its degrees in descending order\n");
  for(i=0;i<n;i++)
    {
      insert(&l2,&r2);
    }
  set_value(&r1);
  set_value(&r2);
  printf("The polynomials entered are : \n");
  display(l1);
  printf("\n");
  display(l2);
  printf("\nThe sum of the polynomials is : ");
  calc(l1,l2);
  printf("%d^x%d \n",(r1->coef)+(r2->coef),r1->power);
}

void insert(nodetype **l,nodetype **r)
{
  nodetype *p=NULL;
  int num=0,pow=0;
  p=(nodetype*)malloc(sizeof(nodetype));
  printf("Enter the coefficient : ");
  scanf("%d",&num);
  printf("Enter the power : ");
  scanf("%d",&pow);
  if(p!=NULL)
```

```
    {
        p->coef=num;
        p->power=pow;
        if(*r==NULL && *l==NULL)
        {
            (*r)=(*l)=p;
        }
        else
        {
            (*r)->next=p;
            (*r)=p;
        }
        (*r)->next=NULL;
    }
}


void display(nodetype *l)
{
    while(l!=NULL)
    {
        printf("%dx^%d ",l->coef,l->power);
        l=l->next;
    }
}


void calc(nodetype *l1,nodetype *l2)
{
    while(l1->power!=0 || l2->power!=0)
    {
        if(l1->power>l2->power)
        {
            printf("%dx^%d",l1->coef,l1->power);
            l1=l1->next;
        }
        else if(l1->power==l2->power)
        {
            printf("%dx^%d",(l1->coef)+(l2->coef),l2->power);
            l1=l1->next;
            l2=l2->next;
        }
        else if(l1->power<l2->power)
        {
            printf("%dx^%d",l2->coef,l2->power);
            l2=l2->next;
        }
        printf(" + ");
    }
}


void set_value(nodetype **r)
{
    if((*r)->power==0)
    {
        return;
    }
    else
    {
        nodetype *p=NULL;
        p=(nodetype*)malloc(sizeof(nodetype));
        p->power=0;
        p->coef=0;
        (*r)->next=p;
        (*r)=p;
        (*r)->next=NULL;
    }
}
```



```
Enter the number of terms in the first polynomial : 3
Enter the 1st polynomial with its degrees in descending order
Enter the coefficient : 2
Enter the power : 2
Enter the coefficient : 4
Enter the power : 1
Enter the coefficient : 5
Enter the power : 0

Enter the number of terms in the second polynomial : 2
Enter the 2nd polynomial with its degrees in descending order
Enter the coefficient : 2
Enter the power : 2
Enter the coefficient : 1
Enter the power : 1
The polynomials entered are :
2x^2 4x^1 5x^0
2x^2 1x^1 0x^0
The sum of the polynomials is : 4x^2 + 5x^1 + 5^x0
```