

WEEK 8 - Implementation of Binary Search Tree

```
#include <stdio.h>

#include <stdlib.h>

struct Tree
{
    int ele;
    struct Tree *left;
    struct Tree *right;};

typedef struct Tree tree;

//tree *root=NULL;

tree *create(tree *root,int x)
{
    if(root==NULL)
    {
        tree *newnode=malloc(sizeof(tree));
        newnode->ele=x;
        newnode->left=NULL;
        newnode->right=NULL;
        root=newnode;}
    else if(x<root->ele)
    {
        root->left=create(root->left,x);
    }
    else if(x>root->ele)
    {
        root->right=create(root->right,x);
    }
    return root;
}

void inorder(tree *root)
{
    if(root!=NULL)
    {
        inorder(root->left);
        printf("%d ",root->ele);
    }
}
```

```

        inorder(root->right);
    }

}

void preorder(tree *root)
{ if(root!=NULL)
  {
    printf("%d ",root->ele);
    preorder(root->left);
    preorder(root->right);
  }

}

void postorder(tree *root)
{ if(root!=NULL)
  {
    postorder(root->left);
    postorder(root->right);
    printf("%d ",root->ele);
  }

}

int main()
{ tree *root=NULL;
  int n,x;
  printf("ENTER NO OF ELEMENTS");
  scanf("%d",&n);
  printf("ENTER THE ELEMENTS ");
  for(int i=0;i<n;i++)
  { scanf("%d",&x);

```

```
    root=create(root,x);  
}  
printf("INORDER TRAVERSAL IS ");  
inorder(root);  
printf("\nPOSTORDER TRAVERSAL IS ");  
postorder(root);  
printf("\nPREORDER TRAVERSAL IS ");  
preorder(root);  
  
return 0;  
}
```

OUTPUT:-

ENTER NO OF ELEMENTS7

ENTER THE ELEMENTS 100 90 110 80 95 105 111

INORDER TRAVERSAL IS 80 90 95 100 105 110 111

POSTORDER TRAVERSAL IS 80 95 90 105 111 110 100

PREORDER TRAVERSAL IS 100 90 80 95 110 105 111