week-10

```
#include<stdio.h>
#include<malloc.h>
typedef struct Node{
 int data;
 struct Node *left, *right;
}node;
node *NewNode(int data){
  node *temp=(node*)malloc(size of(node));
  temp->data=data;
  temp->left=temp->right=NULL;
  return temp;
}
void inorder(mode * root){
  if(root! =NULL){
   inorder(root->left);
   printf("%d ", root->data);
    inorder(root->right);
  }
}
void preorder(node *root){
  if(root! =NULL){
 printf("%d", root->data);
  inorder(root->left);
  inorder(root->right);
 }
}
 void postorder(node *root){
 if(root! =NULL){
   inorder (root ->left);
   inordar (root -> right);
    Printf("%d", root -> data);
  }
}

node *insert (node *node, int data){
  if (node == NULL)
 return NewNode (data);
 if (data <node->data);
 if (data <node -> data)
   node->left =insert (node->left, data);
 else
  node->right=insert(node->right, data);
  return node;
}
 int main(void){
   node *root = NULL;
   root = insert(root,20);
   root = insert (root, 10
   root = insert (root, 12
  root = insert (root, 9
  root = insert (root, 16
  root = insert (root, 8
  root = insert (root, 28
  root =insert (root,100);
  root = irrest (root, 2000);
 Printf(" Inorder traversed :\n");
  inorder(root);
printf("\n\npreorder traversal:\n");
 preorder (root);
printf("\n\npostorder traversal:\n");
 postorder(root);
}
```

Scanned with CamScanner