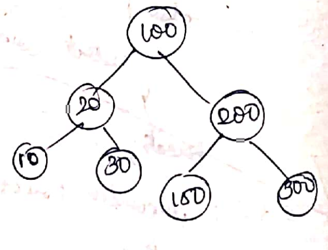
Week -8

19/02/2023

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

struct node{
        int data;
        struct node * left;
        struct node * right;
    };

struct node * create ()
{
  struct node *newnode;
   newnode = (struct node *) malloc(sizeof(struct node));
   pf(" Enter data ");
   sf(" %d", & newnode→ data);
   newnode→ left = newnode →right = NULL ;
    return newnode ;
}.

struct node * insert (struct node * root , struct node *newnode
{ if (root != NULL)
  { if (root →data < newnode →data)
     { insert (root→ Left, newnode);
     }
    else
     {
      root →left = newnode;
     }
  }.
if (root != NULL){
  if (root ->data > newnode-> data){
       insert (root-> ~~left~~ right ;newnode );
     }
  else {
       root -> right = newnode ;
     }
   }
  return root;
}

struct node * preorder (struct node *root)
{ if(root != NULL){
       printf("\t %d" ,root ->data);
       preorder (root->left);
       preorder (root ->right);
     }
}

struct node * inorder (struct node * root )
 {
   if (root != NULL) {
      postorder (root -> left );
      printf(" \t %d", root ->data);
      inorder ( root-> right);
 } }
```

```
struct node *postorder (struct node *root)
{ if (root  !=NULL){
      postorder (root->left);
      printf("\t %d", root->data);
      postorder (root -> right);
  }
}.
int main(){
   int option;
   struct node *root = NULL,*  newnode ;
   while (1)
   {
    printf ("\n 1.create . 2. preorder  3.inorder  4. postorder
              5. Exit ");
    printf (" \n Enter option");
    scanf(" %d", & option);
    switch (option)
    {
     case 1 : newnode = create ();
              if (root ==NULL)
                {root = newnode ;
                }
              else {
                 insert (root, newnode );
              }
           break;
     case 2 : preorder (root);
           break ;
     case 3: inorder (root);
           break;
case 4 : postorder (root);
    break;
case 5 : exit(0);
default :
    printf(" wrong ~~value~~ input given: -");
    break;
  }
 }
}
```

```
o/p ⇒
  1.create 2 preorder 3.inorder 4.postorder 5.exit.
Enter option : 1
Enter data : 200.

  1. create 2.preorder 3. inorder 4. postorder 5.exit
Enter option : 1
Enter data : 150.
  1. create 2. preorder 3. inorder 4. postorder 5.exit
Enter option : 1
 Enter data : 300
  1. create 2. preorder 3. inorder 4. postorder. 5.exit.
Enter option : 1
 Enter data : 20
 Enter data : 10
 Enter data : 30
```



```
Enter option : 2
    100, 20, 10, 30, 200, 150, 300.

Enter option : 3 .
    10, 20, 30, 100, 150, 200, 300

Enter option : 4
    10, 30, 20, 200, 150, 300, 100.
```

19-02-24