IBM19CS216.

Lab: 10

WAP to implement Binary Search tree:
a) to construct a binary search tree.
b) to traverse the tree using all methods (in-order, pre-order & post order).
c) To display all the elements in the tree.

```
struct node
{
    int info;
    struct node *lchild;
    struct node *rchild;
} *root;

void insert (int item)
{
    struct node *tmp, *parent, *location;
    find (item, &parent, &location);
    if (location != NULL)
    {
    printf ("Item already present");
    return;
    }
tmp = (struct node *) malloc (size of (struct node));
tmp->info = item;
tmp->lchild = NULL;
tmp->rchild = NULL;
```

```
if (parent == NULL)
    root=tmp;
else
  if (item->parent->info)
    parent->lchild =tmp;
  else
    parent->rchild =tmp;

int preorder (struct node *ptr)
{
  if (root == NULL)
  {
    printf ("Tree is empty");
  }
  if (ptr != NULL)
  {
    printf ("%d", ptr->info);
    preorder (ptr->lchild);
    preorder (ptr->rchild);
  }
}

void inorder (struct node *ptr)
{
  if (root == NULL)
  {
    printf ("Tree is empty");
  }
```

```
if (ptr != NULL)
{
    inorder (ptr->lchild);
    printf ("%d", ptr->info);
    inorder (ptr->rchild);
}

void postorder (struct node *ptr)
{
    if (root == NULL)
    {
        printf ("Tree is empty");
    }
if (ptr != NULL)
{
    postorder (ptr->lchild);
    postorder (ptr->rchild);
    printf ("%d", ptr->info);
}
}
```