LAB-9

```
struct node
{
  int key;
  struct node * left;
  struct node * right;
}

 struct node* create(int data)
 {
     struct node * temp;
     temp=( struct -node *) malloc (sizeof (struct node));
      temp -> key = data;
      temp -> left = temp -> right =NULL;
      return temp;
   }
   void insert( struct node *root, struct node * temp)
   {
     if (temp -> key < root -> key)
     {
        if (root -> left ! =NULL)
             insert(root -> left, temp);
        else
             root -> left =temp;
       }
      if (temp -> key > root -> key)
      {
          if (root -> right ! =NULL)
            insert( root -> right , temp);
           else
           root -> right = temp;
      } }
```

```
void    inorder ( struct  node * root)
 {
        if ( root != NULL)

     {
         inorder (root -> left );
         printf(" %d", root -> key).
           inorder (root -> right);
        }
      }
   void    preorder( struct  node * root)
   {
         if  (root != NULL)
           {
                printf ("%d", root -> key);
                preorder (root -> left );
                 preorder ( root -> right );
           }
        }
    void   postorder (struct  node *  root)
    {
           if  (root != NULL)
            {
                postorder (root -> left);
                 postorder ( root -> right);
                printf( "%d", root -> key);
             }
         }
       int    main ()
       {
          char ch;
         struct   node *  root  = NULL, * temp;
do {
    temp = create(data);
    if (root ==NULL)
          root=temp;

    else
        insert(root, temp);
    printf("Do yo want to enter more (Y/N)? ");
    getchar();
     scanf("%c", &ch);
  } while (ch=='y" || ch=='Y');
  return 0;
}
```