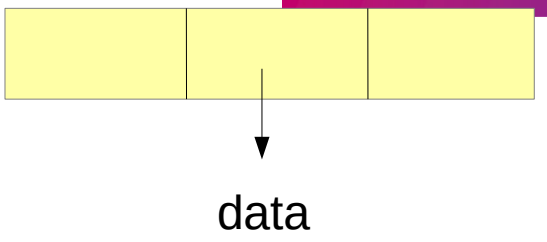
Data Structures
Tree – Binary Search Tree

Team Emertxe

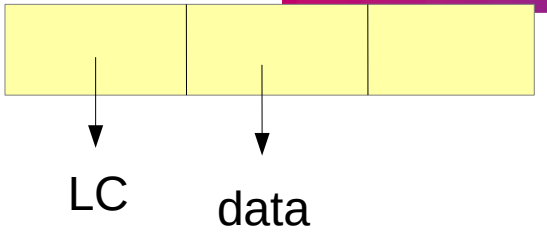
Binary Search Tree - Insert

Insert

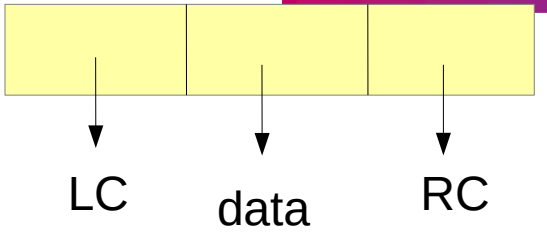
Insert



Insert

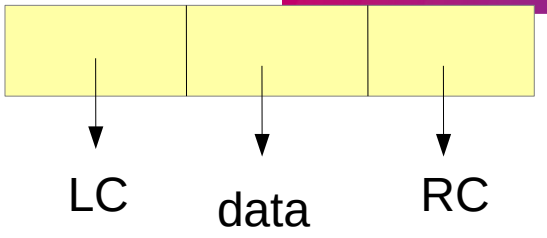


Insert



Insert

```
typedef struct node
{
        struct node * LC;
        int data;
        struct node  *RC;
} tree_t;
```



Address of Left child

Item

Address of right child

Algorithm

bst_insert(root,data):

Input Specification:

root : Pointer that contains address of structure pointer (tree_t)

data : Item to be added

Algorithm

bst_insert(root,data):

Input Specification:

root : Pointer that contains address of structure pointer (tree_t)

data : Item to be added

Output Specification:

Status : e_true / e_false

Data Structure – Binary Search Tree

Algorithm

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

EMERTXE

Data Structure – Binary search Tree

bst_insert(root,data)

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52

EMERTXE

Data Structure – Binary search Tree

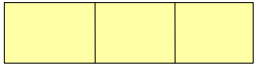
bst_insert(root,data)

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52

new



EMERTXE

Data Structure – Binary search Tree

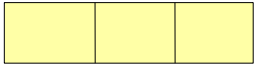
bst_insert(root,data)

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52

new



EMERTXE

Data Structure – Binary search Tree

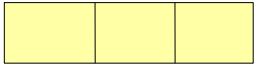
bst_insert(root,data)

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52

new



EMERTXE

Data Structure – Binary search Tree

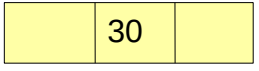
bst_insert(root,data)

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52

new



EMERTXE

Data Structure – Binary search Tree

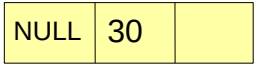
bst_insert(root,data)

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52

new



EMERTXE

Data Structure – Binary search Tree

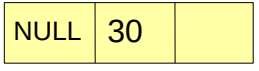
bst_insert(root,data)

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52

new



EMERTXE

Data Structure – Binary search Tree

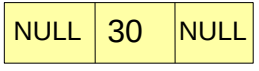
bst_insert(root,data)

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52

new



EMERTXE

Data Structure – Binary search Tree

bst_insert(root,data)

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52

new

NULL	30	NULL

EMERTXE

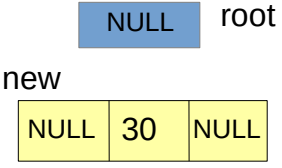
Data Structure – Binary search Tree

bst_insert(root,data)

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52



EMERTXE

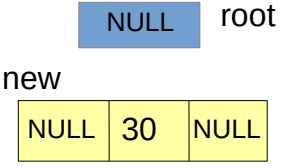
Data Structure – Binary search Tree

bst_insert(root,data)

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52



EMERTXE

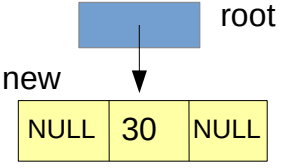
Data Structure – Binary search Tree

bst_insert(root,data)

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52



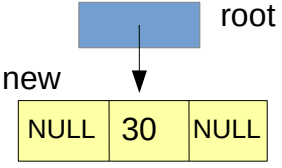
EMERTXE

Data Structure – Binary search Tree

bst_insert(root,data)

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52



EMERTXE

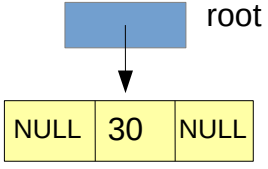
Data Structure – Binary search Tree

bst_insert(root,data)

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52



new

EMERTXE

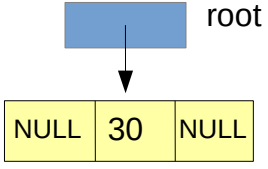
Data Structure – Binary search Tree

bst_insert(root,data)

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52



new

EMERTXE

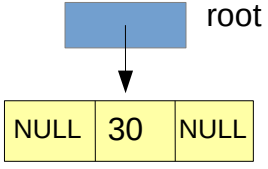
Data Structure – Binary search Tree

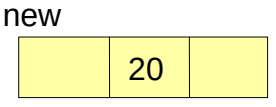
bst_insert(root,data)

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52





EMERTXE

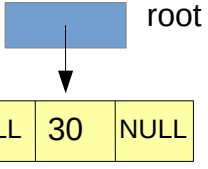
Data Structure – Binary search Tree

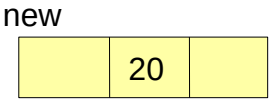
bst_insert(root,data)

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52





EMERTXE

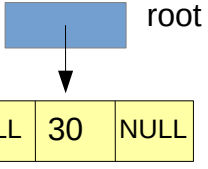
Data Structure – Binary search Tree

bst_insert(root,data)

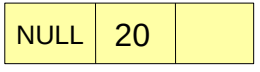
```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52



new



EMERTXE

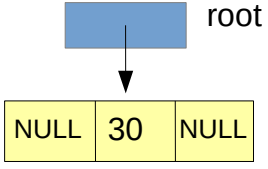
Data Structure – Binary search Tree

bst_insert(root,data)

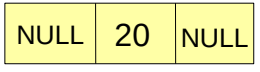
```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52



new



EMERTXE

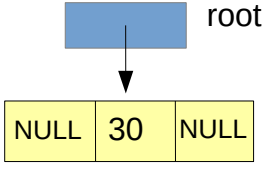
Data Structure – Binary search Tree

bst_insert(root,data)

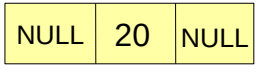
```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52



new



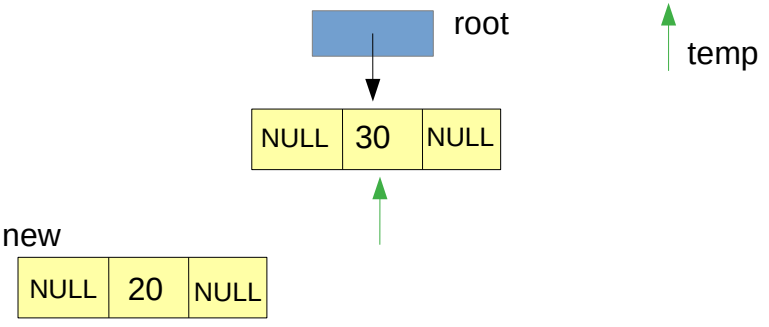
Data Structure – Binary search Tree

bst_insert(root,data)

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52



EMERTXE

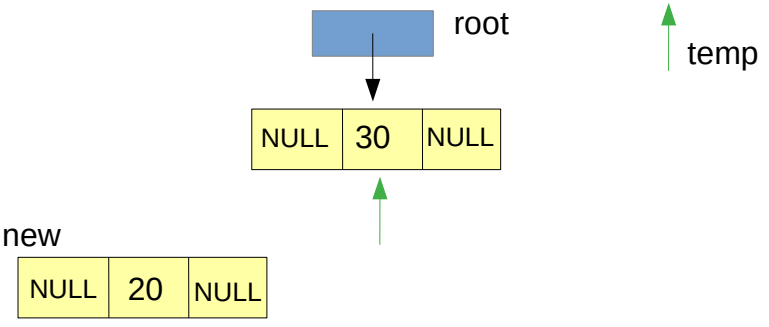
Data Structure – Binary search Tree

bst_insert(root,data)

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52



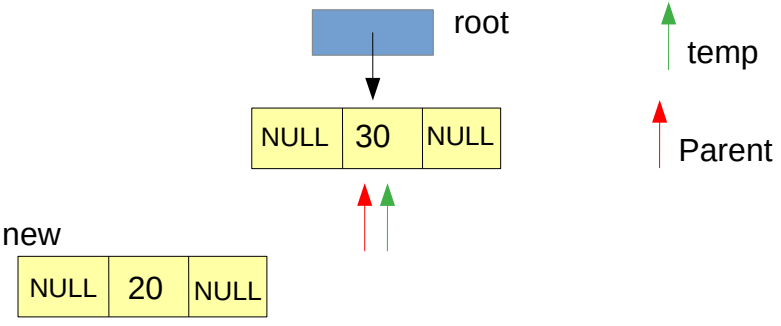
EMERTXE

Data Structure – Binary search Tree

bst_insert(root,data)

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52



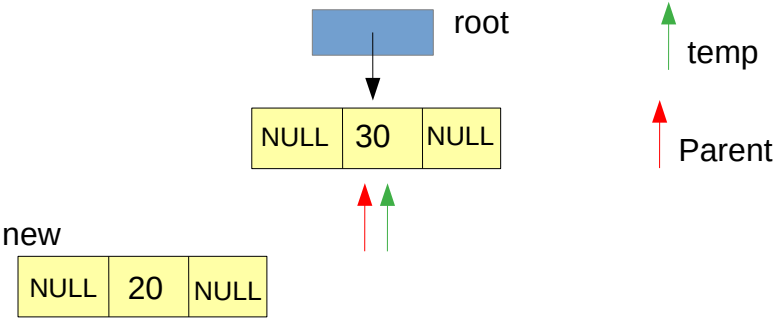
EMERTXE

Data Structure – Binary search Tree

bst_insert(root,data)

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52



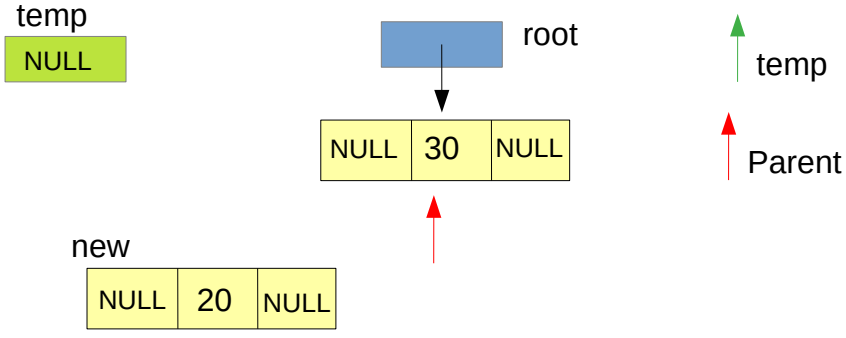
EMERTXE

Data Structure – Binary search Tree

bst_insert(root,data)

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52



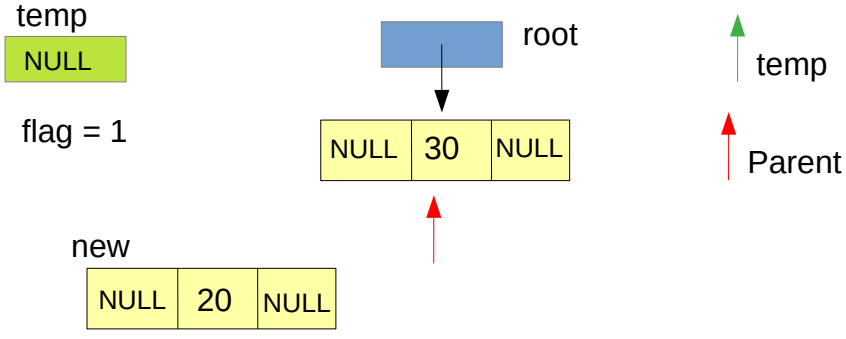
EMERTXE

Data Structure – Binary search Tree

bst_insert(root,data)

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52



EMERTXE

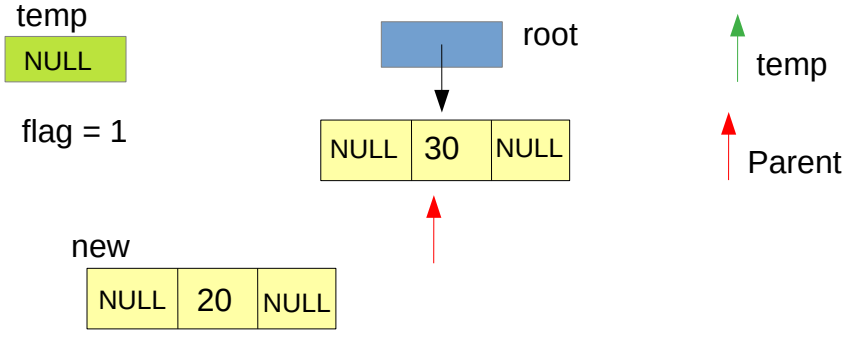
Data Structure – Binary search Tree

bst_insert(root,data)

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52



EMERTXE

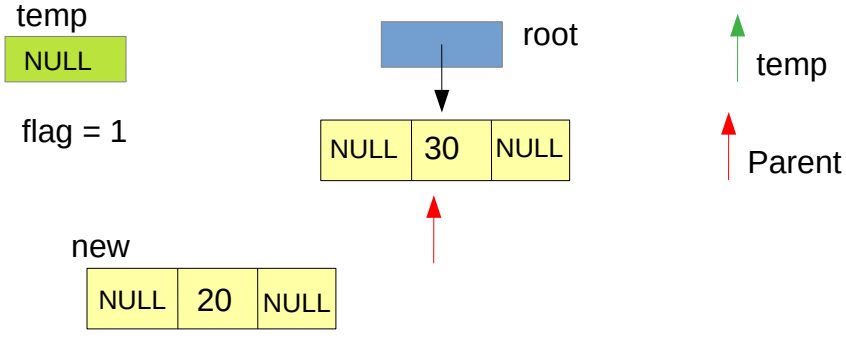
Data Structure – Binary search Tree

bst_insert(root,data)

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52



EMERTXE

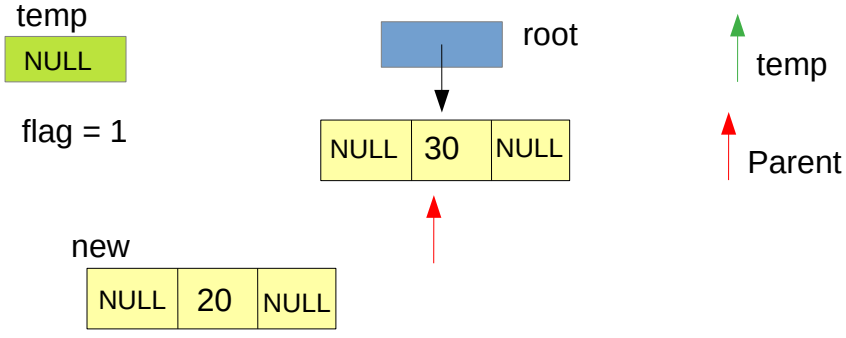
Data Structure – Binary search Tree

bst_insert(root,data)

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52



EMERTXE

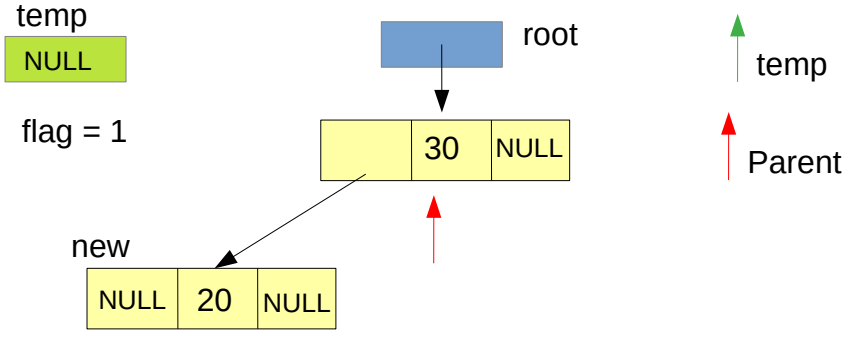
Data Structure – Binary search Tree

bst_insert(root,data)

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52



EMERTXE

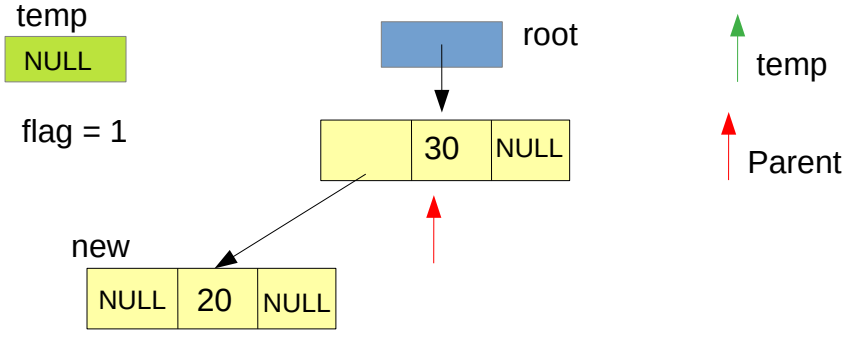
Data Structure – Binary search Tree

bst_insert(root,data)

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52



EMERTXE

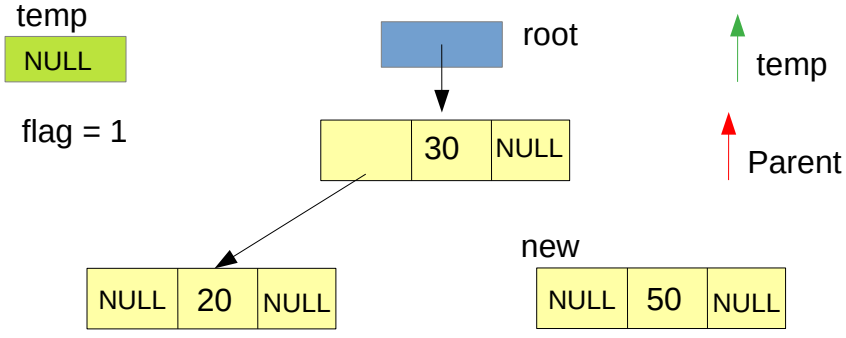
Data Structure – Binary search Tree

bst_insert(root,data)

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52



EMERTXE

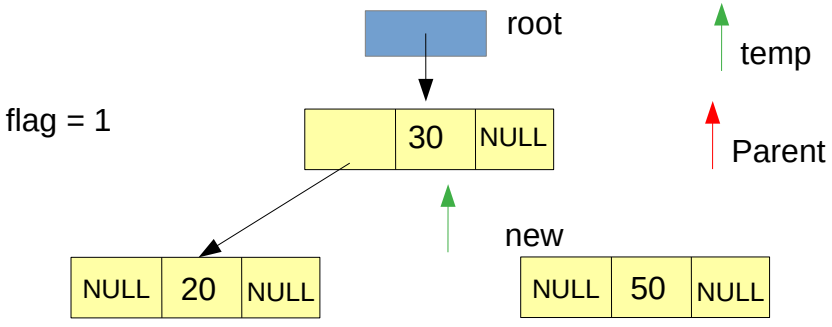
Data Structure – Binary search Tree

bst_insert(root,data)

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52



EMERTXE

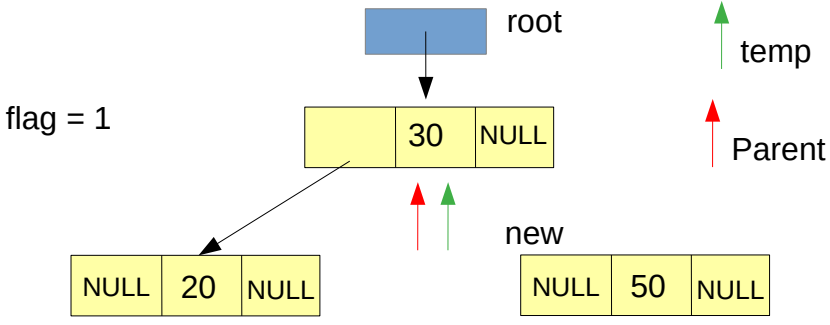
Data Structure – Binary search Tree

bst_insert(root,data)

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52



EMERTXE

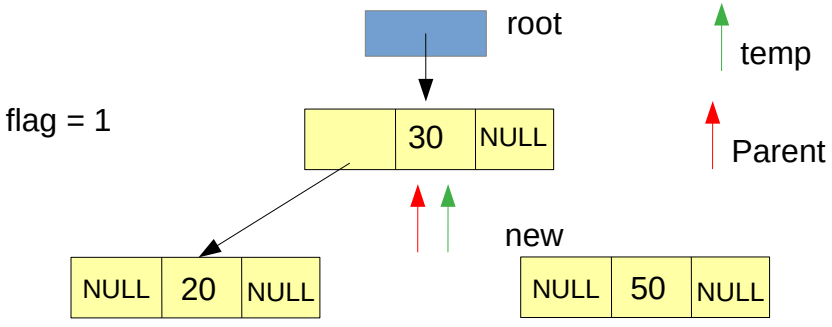
Data Structure – Binary search Tree

bst_insert(root,data)

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52



EMERTXE

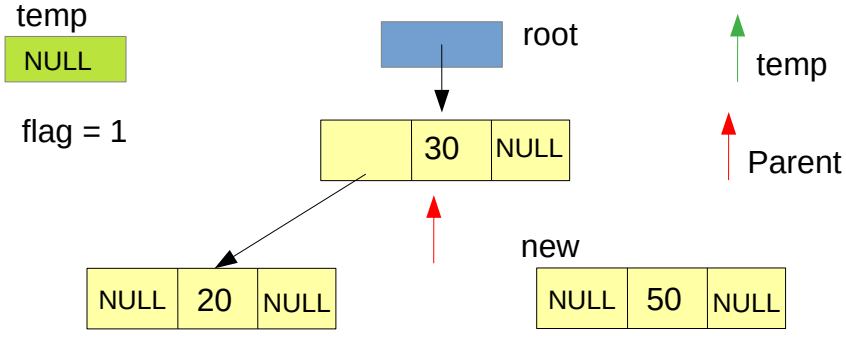
Data Structure – Binary search Tree

bst_insert(root,data)

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52



EMERTXE

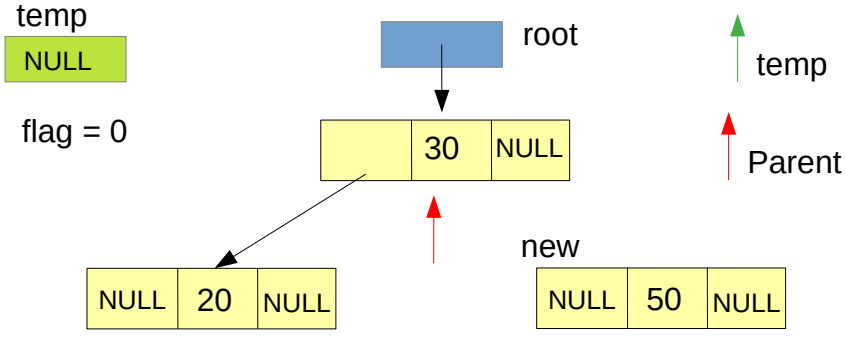
Data Structure – Binary search Tree

bst_insert(root,data)

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52



EMERTXE

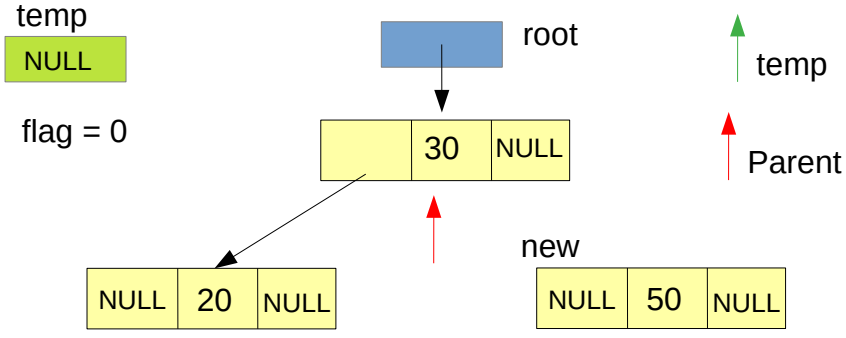
Data Structure – Binary search Tree

bst_insert(root,data)

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52



EMERTXE

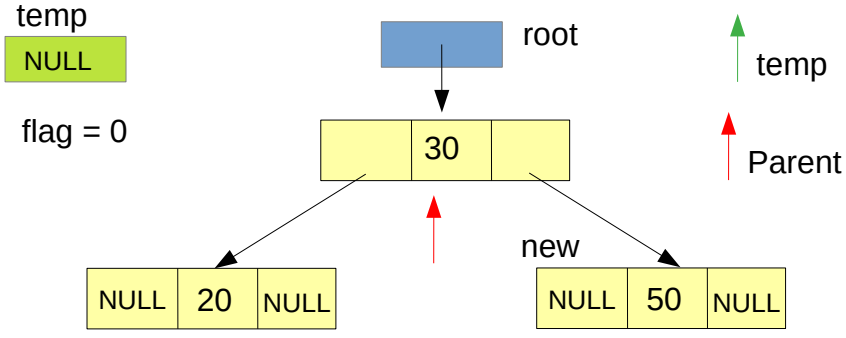
Data Structure – Binary search Tree

bst_insert(root,data)

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52



EMERTXE

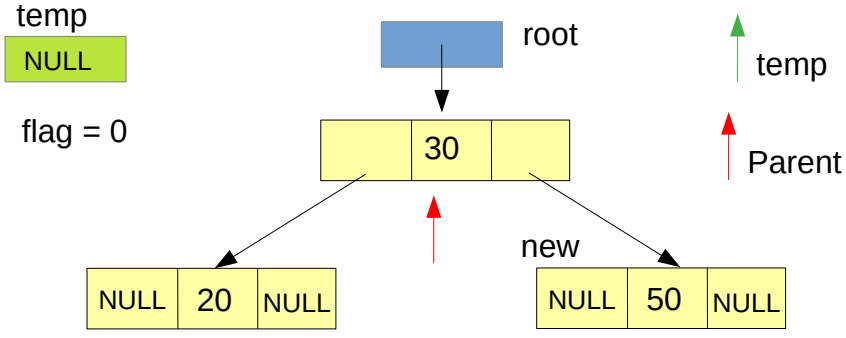
Data Structure – Binary search Tree

bst_insert(root,data)

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52



EMERTXE

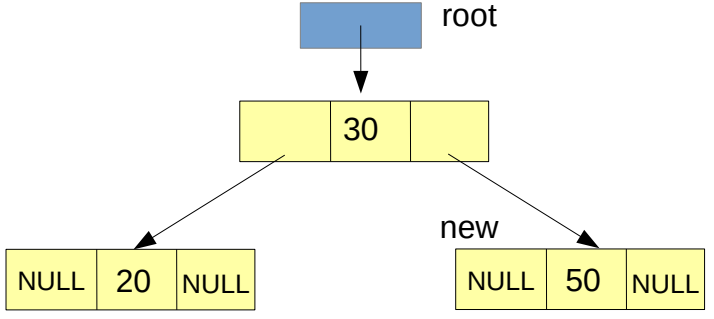
Data Structure – Binary search Tree

bst_insert(root,data)

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52



EMERTXE

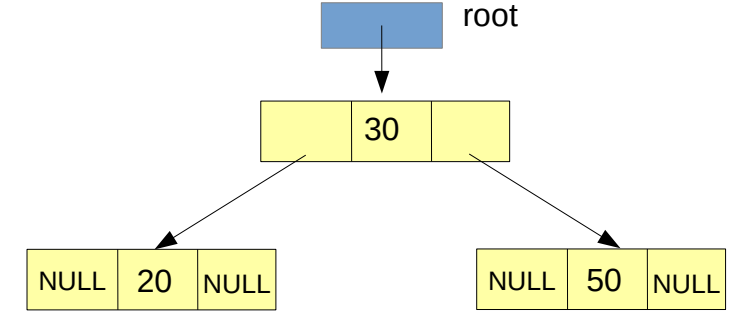
Data Structure – Binary search Tree

bst_insert(root,data)

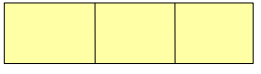
```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52



new



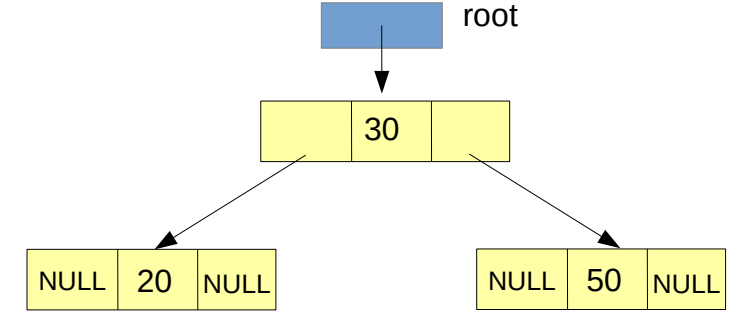
Data Structure – Binary search Tree

bst_insert(root,data)

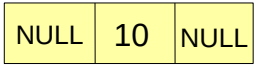
```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52



new



EMERTXE

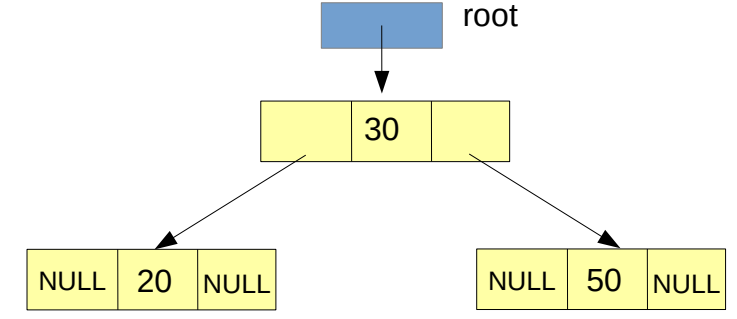
Data Structure – Binary search Tree

bst_insert(root,data)

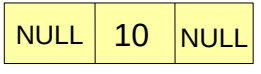
```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52



new



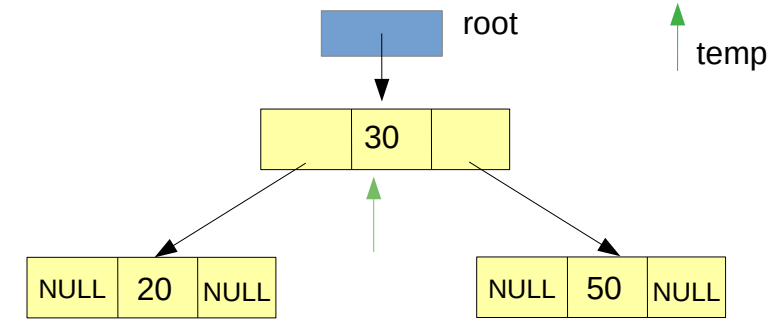
Data Structure – Binary search Tree

bst_insert(root,data)

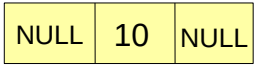
```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52



new



EMERTXE

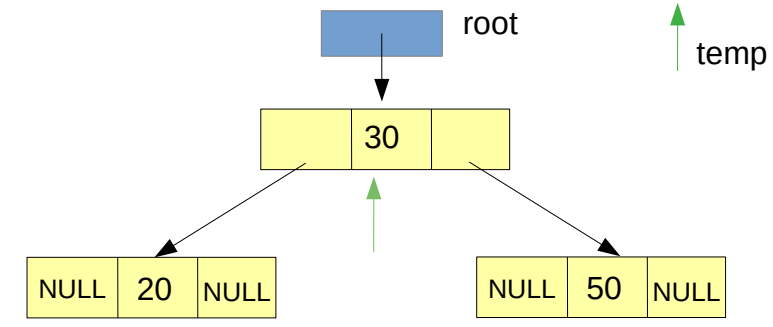
Data Structure – Binary search Tree

bst_insert(root,data)

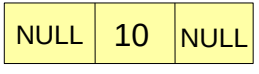
```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52



new



EMERTXE

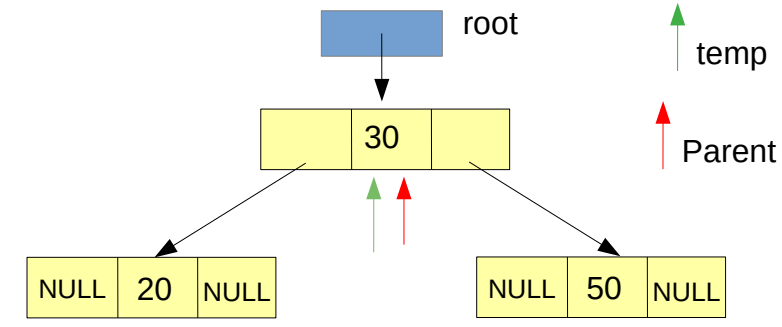
Data Structure – Binary search Tree

bst_insert(root,data)

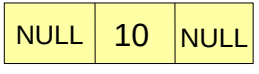
```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52



new



EMERTXE

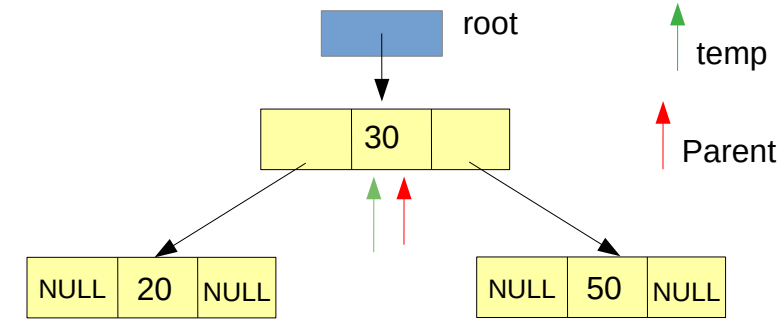
Data Structure – Binary search Tree

bst_insert(root,data)

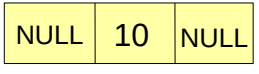
```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52



new



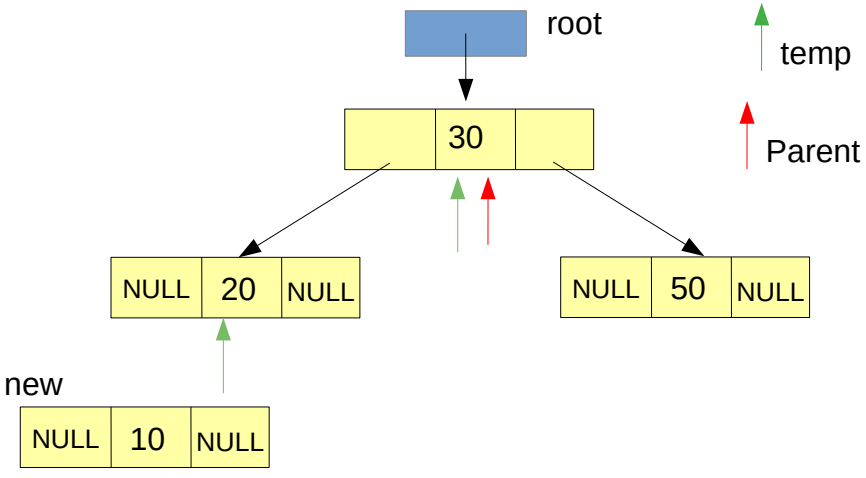
EMERTXE

Data Structure – Binary search Tree

bst_insert(root,data)

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52



EMERTXE

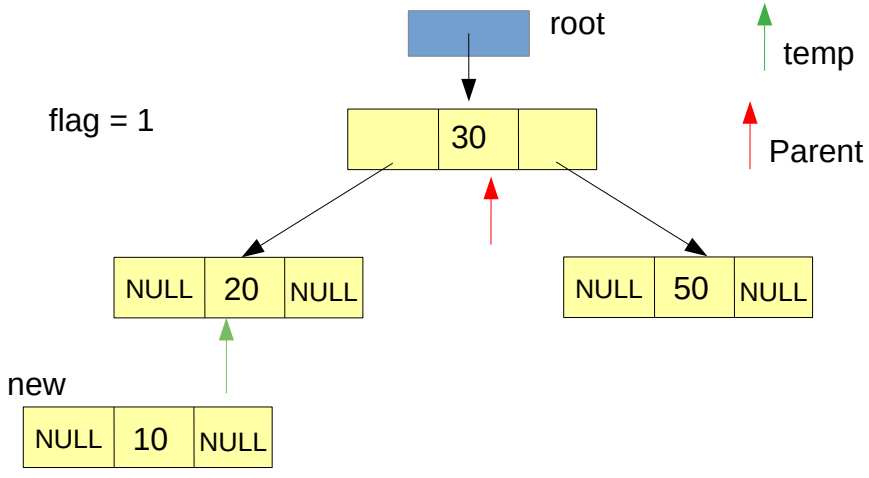
Data Structure – Binary search Tree

bst_insert(root,data)

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52



EMERTXE

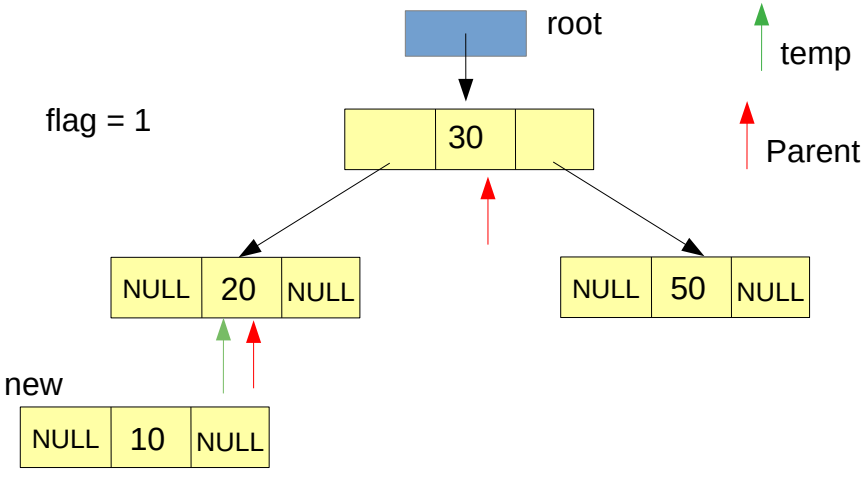
Data Structure – Binary search Tree

bst_insert(root,data)

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52



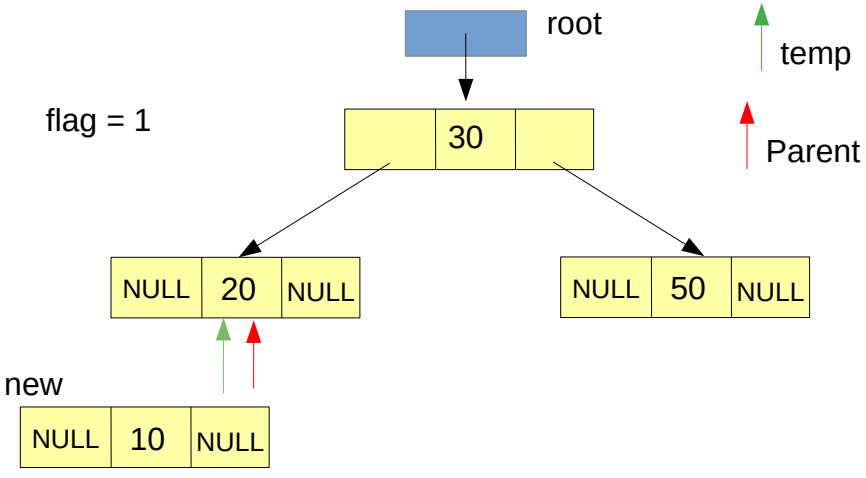
EMERTXE

Data Structure – Binary search Tree

bst_insert(root,data)

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52



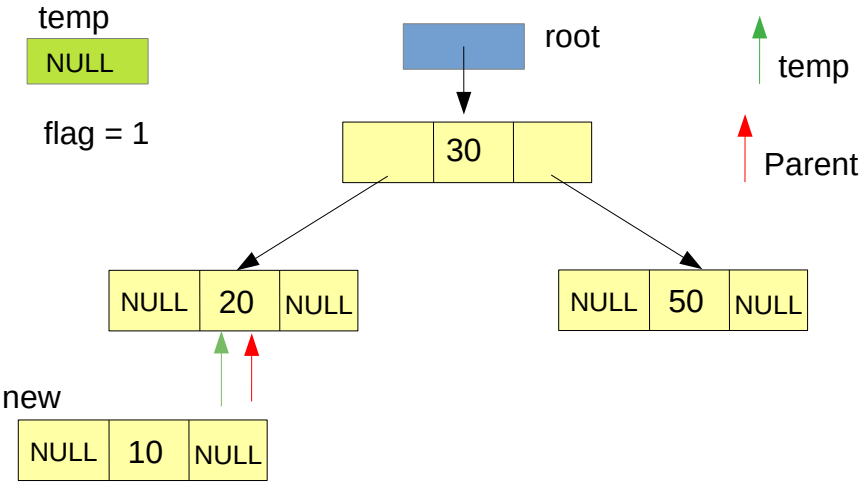
EMERTXE

Data Structure – Binary search Tree

bst_insert(root,data)

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52



EMERTXE

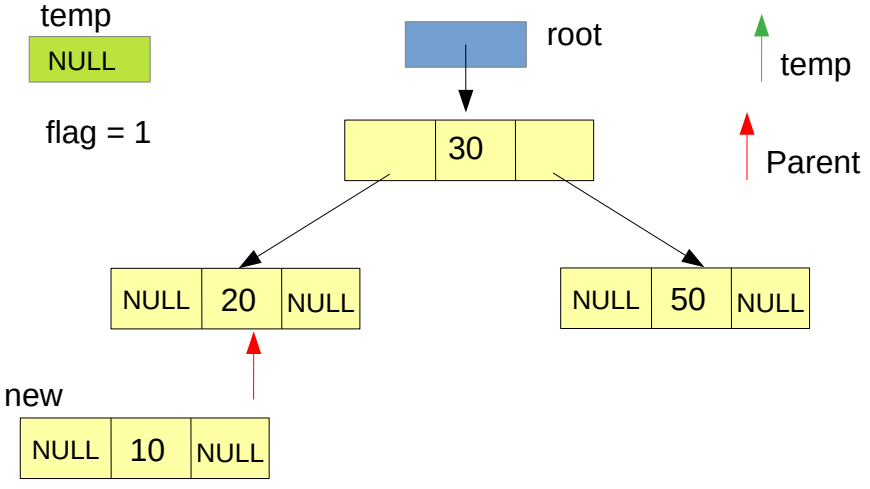
Data Structure – Binary search Tree

bst_insert(root,data)

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52



EMERTXE

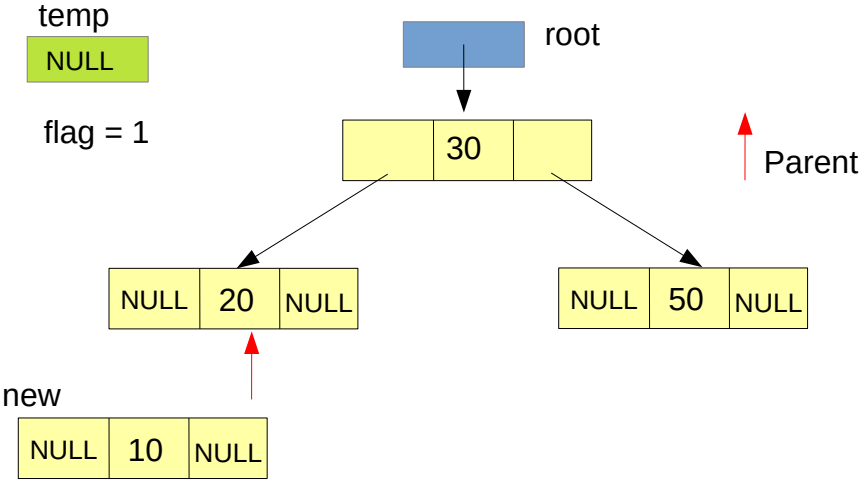
Data Structure – Binary search Tree

bst_insert(root,data)

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
  Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52



EMERTXE

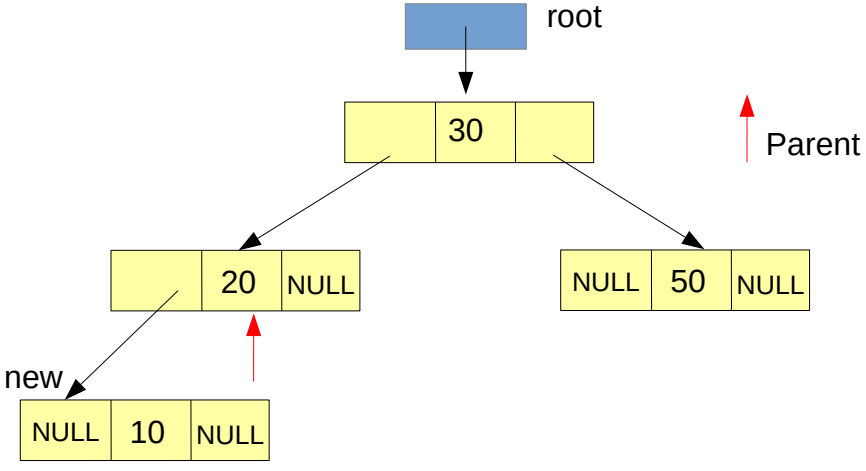
Data Structure – Binary search Tree

bst_insert(root,data)

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52



EMERTXE

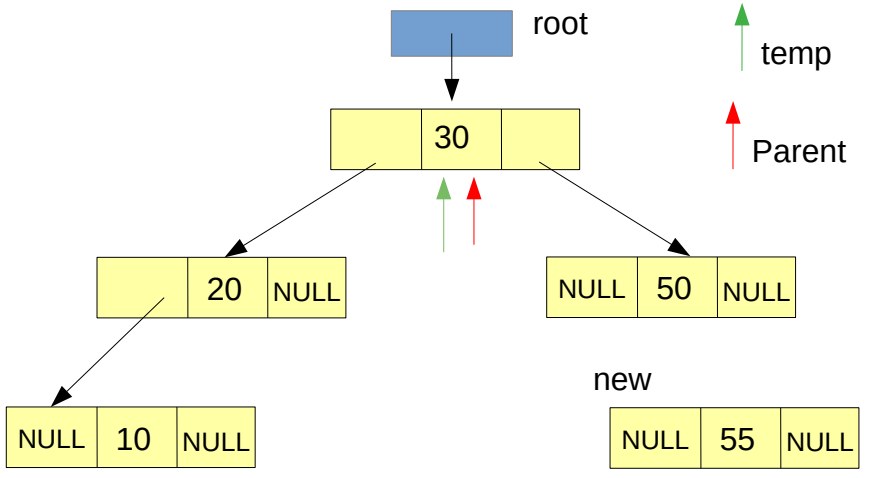
Data Structure – Binary search Tree

bst_insert(root,data)

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52



EMERTXE

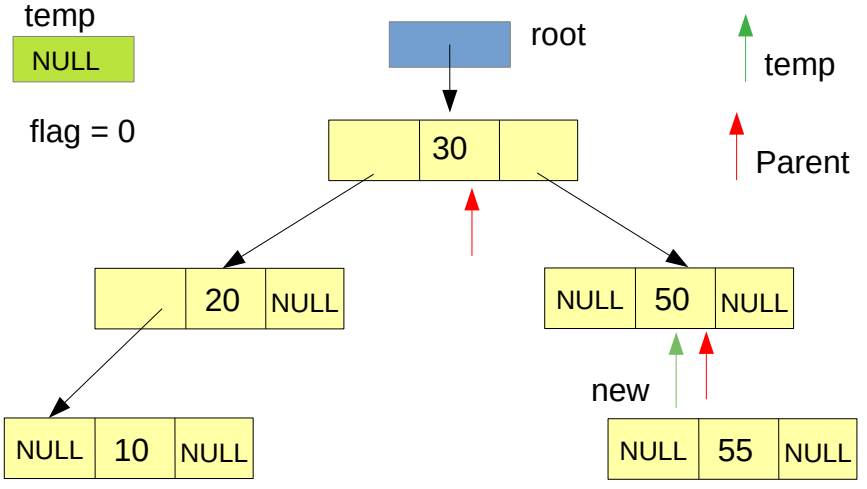
Data Structure – Binary search Tree

bst_insert(root,data)

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52



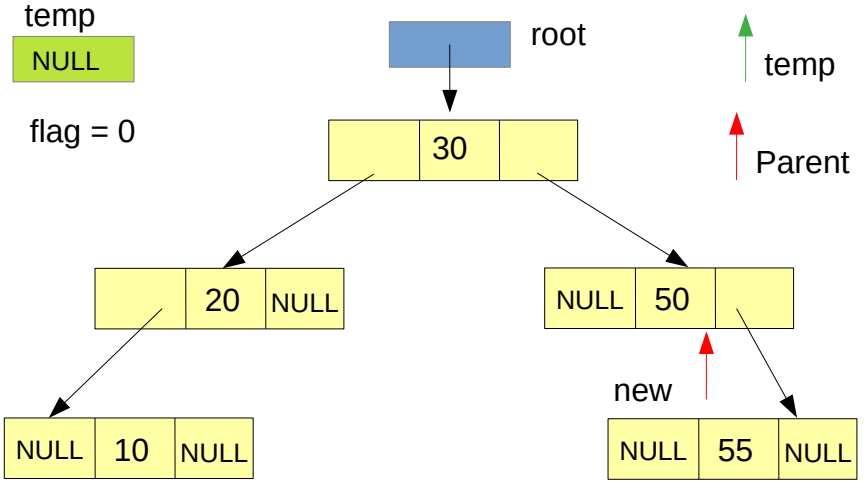
EMERTXE

Data Structure – Binary search Tree

bst_insert(root,data)

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52



EMERTXE

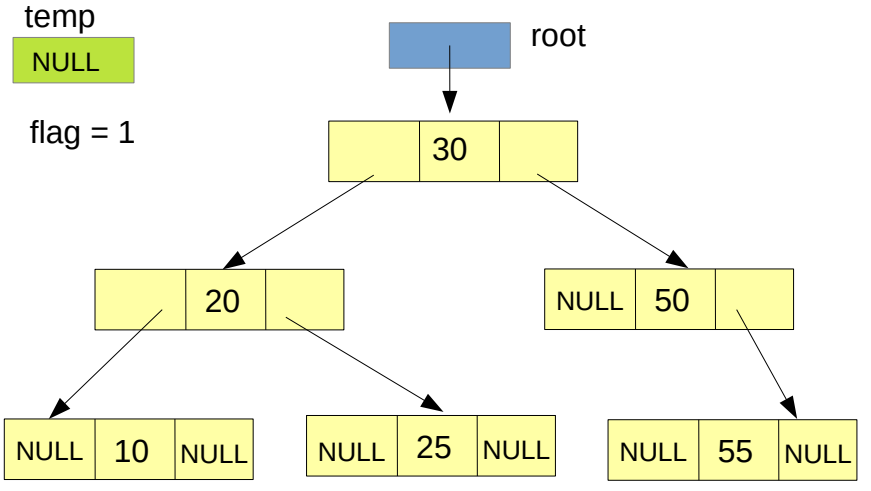
Data Structure – Binary search Tree

bst_insert(root,data)

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52



EMERTXE

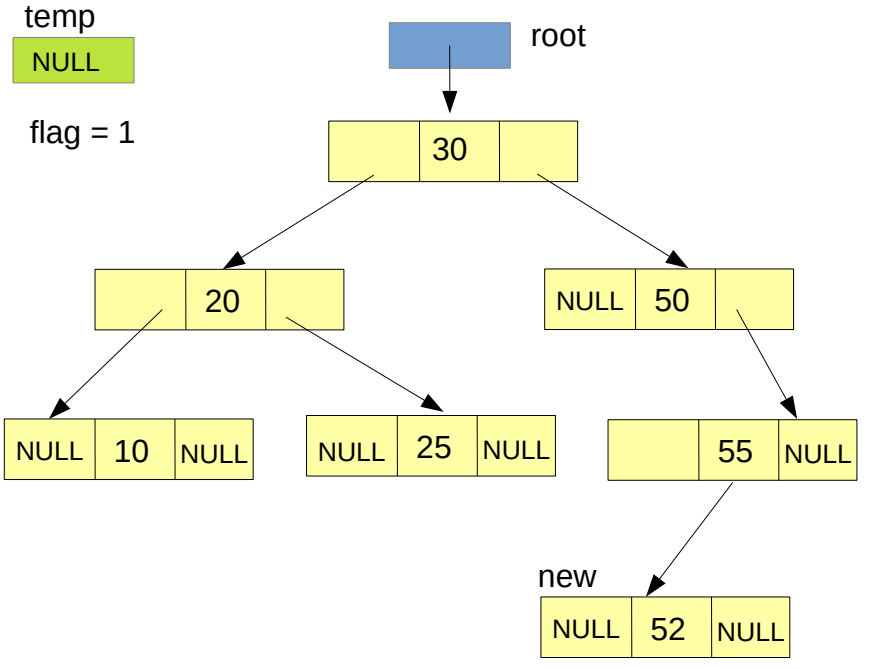
Data Structure – Binary search Tree

bst_insert(root,data)

```
new = Memalloc(sizeof(tree_t))
if(new = NULL)
        return e_false
new->data = data
new ->LC =NULL
new->RC = NULL
if(root = NULL)
   root = new
   return e_true
temp = root
while(temp != NULL)
   Parent = temp
   If( data < temp -> data )
      temp = temp -> LC
      flag = 1
   Else if (data > temp -> data)
        temp = temp ->RC
        flag = 0
   Else
        return DUPLICATE_FOUND
```

```
If (flag = 1)
   Parent ->LC = new
Else
   Parent -> RC = new
return e_true
```

30 , 20 , 50 , 10 , 55, 25, 52



EMERTXE

Code - bst_insert(root,data)