

04/01/2021

DS LABTEST - 2

Q. write a program to construct a BST and also to find the maximum value in a BST.

A. struct node

{

int key;

struct node * left;

struct node * right;

}

struct node * root;

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;

}

}

}

display

void ~~display~~ (struct node *root)

{

if (root != NULL)

{

display(root → left);

printf("a %d", root → key);

display(root → right);

}

}

struct node *Max(struct node *~~root~~)

{

if (root == NULL)

return NULL;

else if (root → right == NULL) return root;

else

return Max(root → right);

}

04/01/21

```
int main()  
{
```

```
    char ch;
```

```
    struct node *root = NULL; *temp;
```

```
    do {
```

```
        temp = create(data);
```

```
        if (root == NULL)
```

```
            root = temp;
```

```
        else
```

```
            insert(root, temp);
```

```
        printf("Enter more (Y/N)? ");
```

```
        getch();
```

```
        scanf("%c", &ch);
```

```
    } while (ch == 'y' || ch == 'Y');
```

```
    return 0;
```

```
}
```