**void DELETE\_NODE\_WITH\_SINGLE\_CHILD()**

**40**

**33**

**11**

**60**

**50**

**45**

**55**

**{**

**BINARY\_S\_TREE \*loc,\*par, \*child;**

**int item;**

**loc=root;**

**printf("\nENTER THE NODE TO BE DELETED ");**

**scanf("%d",&item);**

**while(loc->info != item)**

**{**

**if(item < loc->info)**

**{**

**par=loc;**

**loc=loc->left;**

**}**

**When to Delete Node with Single Child; Simply Replace the Position of That Node With Its Single Child Node**

**else**

**{**

**par=loc;**

**loc=loc->right;**

**40**

**11**

**60**

**50**

**45**

**55**

**}**

**}**

**if(loc->left==NULL)**

**{**

**child=loc->right;**

**}**

**else**

**{**

**child=loc->left;**

**}**

**if(par->info > loc->info)**

**Node Deleted is 33**

**{**

**par->left=child;**

**}**

**else**

**{**

**par->right=child;**

**}**

**loc->right=loc->left=NULL;**

**free (loc);**

**}**