

TI DSP,Xilinx zynq FPGA,MCU 및 Xilinx  
zynq FPGA 프로그래밍 전문가 과정

강사-INNOVA LEE(이상훈)

[Gccompil3r@gmail.com](mailto:Gccompil3r@gmail.com)

학생-윤지완

[Yoonjw7894@naver.com](mailto:Yoonjw7894@naver.com)

## 1.트리 DELETE

```
tree *delete tree(tree* root,int data)
{
    int num;
    tree *tmp;
    if(*root==NULL)
    {
        printf("no fine");
        return NULL;
    }
    else if((root->data>data)
    {
        root->left=delete tree(root->left,data);
    }
    else if((root->data<data)
    {
        root->right=delete tree(root->right,data);
    }
    else if(root->left&&root->right)
    {
        root->left=fine_max(root->left,&num);
        root->data=num;
    }
    else
        root=chg_mode(root);
    return root;
}

void find_max(tree *root,int *data )
{
    if(root->right)
        root->right=fine_max(root->right,data);
    printf("트리의 최상값=%d",root->right);
}
```

```

else
{
    *data=root->data;
    root=chang_mode(root);
}
return root;
}

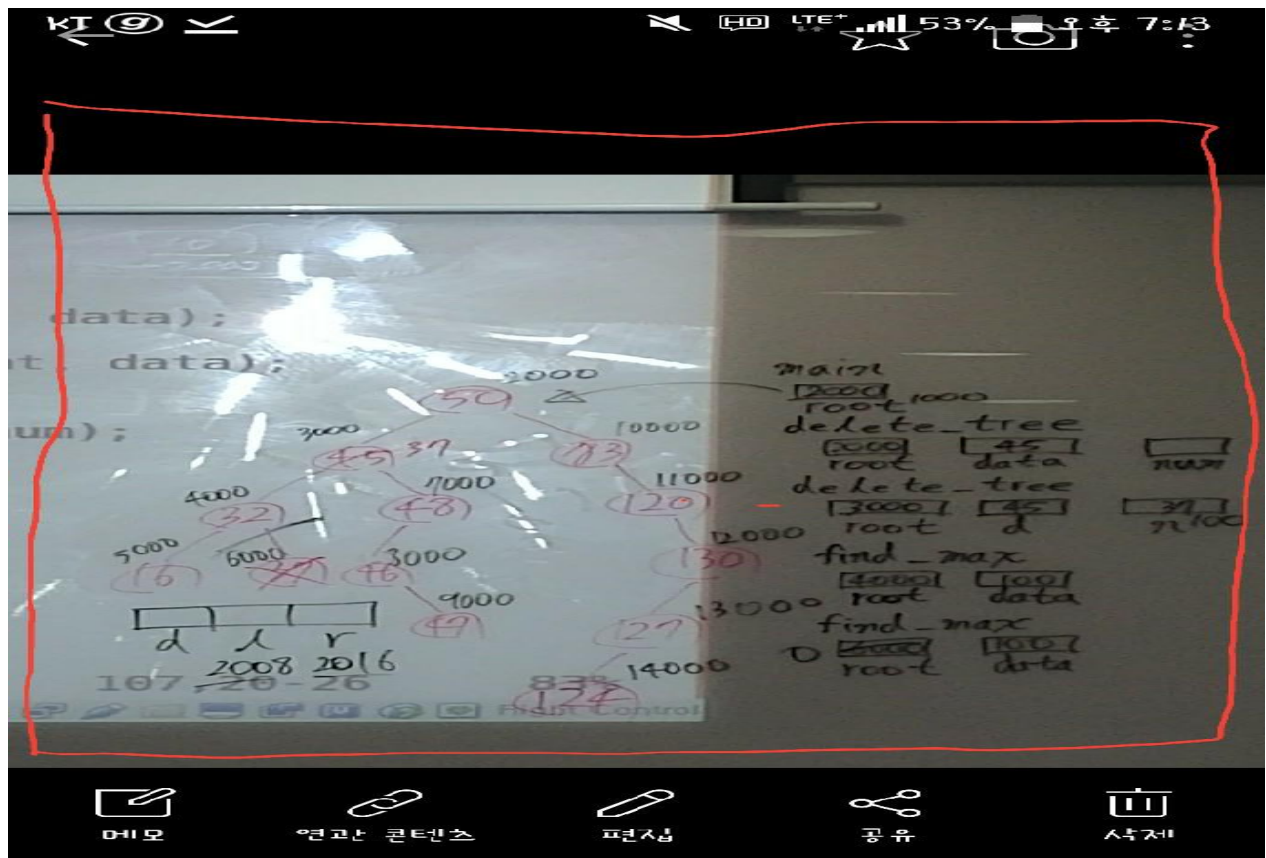
```

```

void *chang_mode(tree *root)
{
    tree *tmp=root;
    if(!root->left)
    {
        root=root->right;
    }
    if(!root->right)
    {
        root=root->left;
    }

    root=root->left;
    free(tmp);
    return root;
}

```



이번에는 tree구조체를 만들고 배열로 값을 left,right포인터로 값을 하고

위에 프로그램을 보면 만약 45를 없앤다고 하면 처음 data값을 45를 넣고 주소가 3000번지가 된다.그럼 저음 if문은 실행되지않고 처음 50과 비교해 45는 값이 낮으니까

```
else if((root->data>data)
{
root->left=delete tree(root->left,data);
```

의 if문이 실행하고 재귀 함수를 호출해 3000번지에서 root->left를 가리키고 있으므로 숫자 32를 가리키고 주소는 4000지이다. 그 다음 마지막 else if문에서

```
else if(root->left&&root->right)
{
root->left=fine_max(root->left,&num);
root->data=num;
}
```

이 부분이 실행이 된다 저 조건이 만족하는 이유는 32에 양쪽다 값이 있기때문이다. 그 다음

root->left=fine\_max(root->left,&num);이 실행이 되어

```
oid find_max(tree *root,int *data )
{
if(root->right)
root->right=fine_max(root->right,data);
}
else
{
*data=root->data;
root=change_mode(root);
}
return root;
}
```

Fine\_max함수를 호출하고 tree \*root는 32를 가리키고 있으며 if문이 실행되어 4000번지에서 root->right가 실행되고 숫자 37을를 가리키며 그 다음 37번지에는 오른쪽 왼쪽 가지가 존재하지 않기에 else로 넘어가서 root->data에 있는 값을 \*data에 넣는다 이 말은 결국 num에 값을 넣겠다는 것이다.그리고 root=change\_mode가 실행이 되고

```
void *chang_mode(tree *root)
{
tree *tmp=root;
if(!root->left)
{
root=root->right;
}
if(!root->right)
{
root=root->left;
}

root=root->left;
free(tmp);
return root;
```

```
}
```

\*tmp에 root를 가리키게 하고 37은 가지가 아예없기에 값은 “0”이고

**Free(tmp)를 통해서 37값을 가지고 있는 6000번지는  
해체가 되어서 끊어진다.그리고 return root;를 통해서  
37이 반환이 된다. 그럼 이 값은 4000번지에 리턴되  
며**

```
else if((root->data>data)
```

```
{
```

```
root->left=delete tree(root->left,data);
```

처음 data를 넣었던 번지 즉 45 값을 가진 3000 번지에 45를 delete 시키고 37을 넣는다.