2018. 3. 6 화 – 10 회차

과정 : TI, DSP, Xilinx Znq FPGA, MCU 기반의 프로그래밍 전문가 과정

Prof. 이상훈
Stu. 정상용

자료구조_3

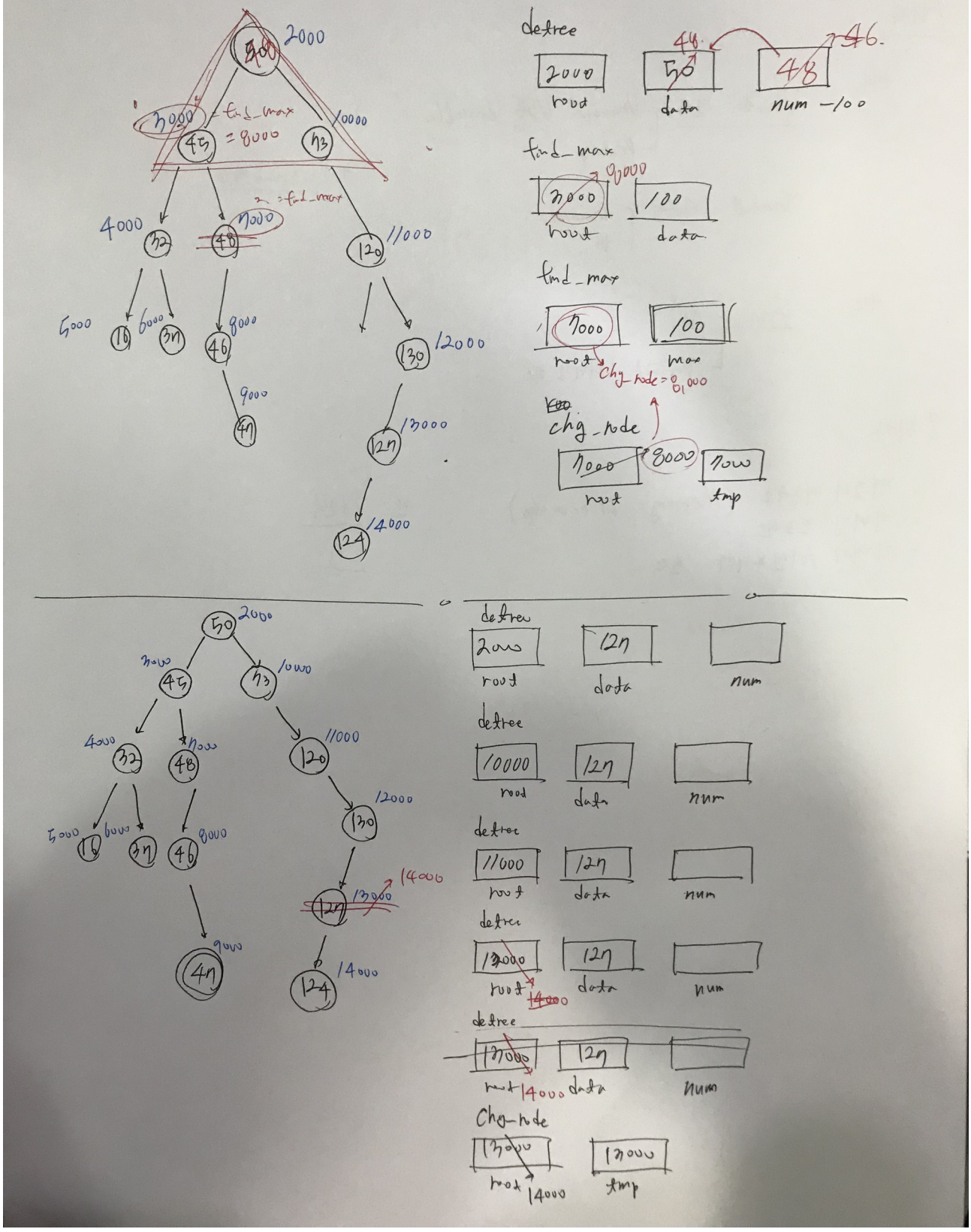
1. Binary_tree

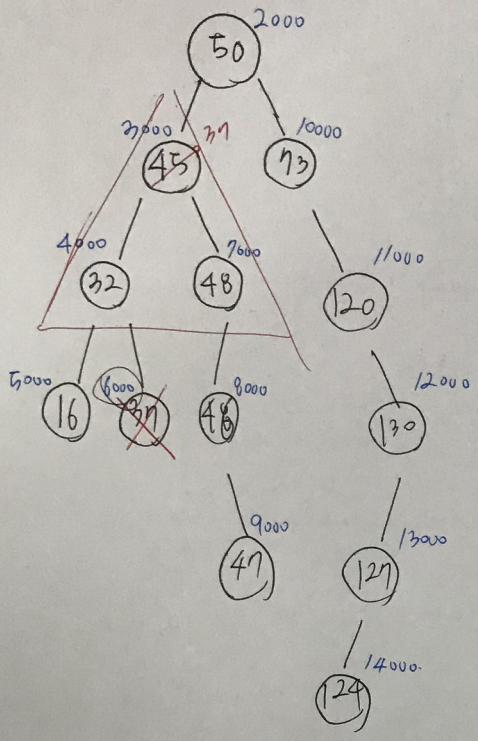
```
#include <stdio.h>
#include <malloc.h>

#define EMPTY 0

typedef struct node
{
   int data;
   struct node *left;
   struct node *right;
}node;

node *get_node()
{
   node *tmp;
   tmp = (node *)malloc(sizeof(node));
   tmp -> left = EMPTY;
   tmp -> right = EMPTY;
   return tmp;
}

void tree_ins(node **root, int data)
{
   if(*root == NULL)
   {
      *root = get_node();
      (*root) -> data = data;
      return ;
   }
   else if(data < (*root)-> data)
   tree_ins(&((*root) -> left), data);
   else if(data > (*root) -> data)
   tree_ins(&((*root) -> right), data);

   return ;
}

int print_tree(node *root)
{
```

```
    if(root)
    {
    printf("%d\n", root -> data);

    print_tree(root -> left);
    print_tree(root -> right);
    }
}

node *chg_node(node *root)
{
   node *tmp = root;

   if(!root -> right)
     root = root -> left;
   else if(!root -> left)
     root = root -> right;

   free(tmp);

   return root;
}


node *find_max(node *root, int *data)
{
   if(root -> right)
       root -> right = find_max(root -> right, data);
   else
   {
       *data = root -> data;
       root = chg_node(root);
   }

   return root;
}

node *detree(node *root, int data)
{
   int num;
   if(root == NULL)
   {
       printf("Not Found\n");
       return NULL;
   }
   else if(root -> data > data)
         root -> left = detree(root -> left, data);

   else if(root -> data < data)
         root -> right = detree(root -> right, data);
```

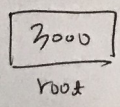
```
    else if(root -> left && root -> right)    //서브트리면
    {
        root -> left = find_max(root -> left, &num);
        root -> data = num;
    }
    else
        root = chg_node(root);
    return root;
}

int main(void)
{
    node *root = EMPTY;
    int i;
    int arr[13] = {50, 45, 73, 32, 48, 46, 16, 37, 120, 47, 130, 127, 124};
    for(i = 0; i < 13; i++)
    {
       tree_ins(&root, arr[i]);
    }
    print_tree(root);

    detree(root, 50);

    print_tree(root);
    return 0;
}
```

Sol)

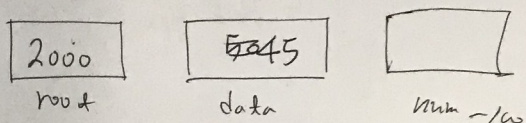

2000
7000
= find_max
= 8000
10000
4000
11000
5000
6000
8000
12000
9000
13000
14000
50
45
13
32
48
120
10
37
46
130
47
127
124
detree
root
data
num
find_max
chg_node
tmp



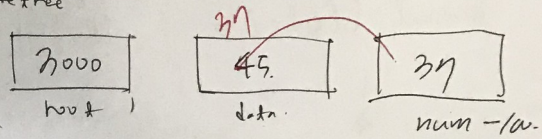
main



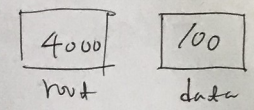
detree



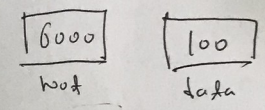
detree



find_max



find_max



chg_node