Assignment-1 (EE4013)

EE18BTECH11013 - Divyansh Maduriya

Download c codes from

https://github.com/Divyansh−28/EE18BTECH11013−EE−4013/tree/master/code

and latex-tikz codes from

https://github.com/Divyansh−28/EE18BTECH11013−EE−4013/tree/master

1 Problem

consider a complete binary tree with 7 nodes. Let A denote the set of first 3 elements obtained by performing Breadth-First search (BFS) starting from root. Let B denote the set of first 3 elements obtained by performing Depth first Search(DFS) starting from the root. the value of

$$|A - B|$$

is ?

2 Solution

Let us start with drawing a 7 node binary tree, then we'll run BFS and DFS from the root node respectively to get set A and B.

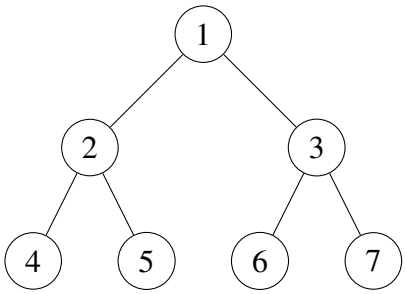

Fig. 0: Binary Tree

Representing the nodes of the binary tree above in the form of an array in the following way.

$$ArrB = [1, 2, 3, 4, 5, 6, 7] \tag{2.0.1}$$

Now, Let's traverse through the array covering the BFS nodes/indexes starting from the root node.

$$i = i + 1 \tag{2.0.2}$$

Where, i represents a particular node/index, starting from i=0(i.e root node).

Now, Let A denote the set of first 3 elements of BFS. Therefore,

$$A = \{1, 2, 3\} \tag{2.0.3}$$

Similarly, Let's traverse through the array covering the DFS nodes/indexes starting from the root node.

$$i = 2i + 1 \tag{2.0.4}$$

Where, i represents a particular node/index, starting from i=0(i.e root node)

Now, Let B denote the set of first 3 elements of DFS. Therefore,

$$B = \{1, 2, 4\} \tag{2.0.5}$$

Now,

$$A - B = \{3\} \tag{2.0.6}$$

and

$$|A - B| = 1 \tag{2.0.7}$$

C Code snippets of the BFS and DFS functions :

BFS C function:

```
    int *bfs(int * tree) {
    int * set = (int* ) malloc(1*sizeof(int));
    int i = 0;
    setbfssz = 0;
    while(setbfssz<3) {

      set = (int *)realloc(set, (setbfssz+1)*sizeof(
          int));
      set[setbfssz] = tree[i];
      i++;
      setbfssz++;
    }
    return set;
}
```

DFS C function:

```
 int * dfs(int * tree) {
    int * set = (int* ) malloc(1*sizeof(int));
    int i = 0;
    setdfssz = 0;
    while(setdfssz < 3) {

      set = (int *)realloc(set, (setdfssz+1)*sizeof(
          int));
      set[setdfssz] = tree[i];
      i = 2*i+1;
      setdfssz++;
    }
    return set;
}
```