LAB # 10 - ASSIGNMENT

24

DEC

2023

**GROUP 16**

MUHAMMAD YASIR | CT-22082

MUHAMMAD SHAHEER QURESHI | CT-22090

SYED SAAD WAQAR | CT-22097

AMMAR YASSER AHMED | CT-22103

NOFIL AHMED KHAN | CT-22301

DATA STRUCTURES ALGORITHMS AND APPLICATIONS (ct – 159)

TEACHER : SIR ABDUL KARIM KAZI

**EXERCISE**

QUESTION 1:

Analyze the given code to find the difference between “find” and “search” methods.

**Find Function:**

**Parameters:**

Find function will take 3 variables in parameters.

1. **First Variable (int I)**  
   One variable will be the value of targeted value which needs to be searched and it is ‘i’.
2. **Second Variable (int \*\*par)**  
   Second variable will be the double pointer variable which will store the address of any other pointer, its name is ‘par’, and at the end of the function, it will be used to store the address of the parent node of the targeted node.
3. **Third Variable (int \*\*loc)**  
   Third variable will be the double pointer variable which will store the address of any other pointer, its name is ‘loc’, and at the end of the function, it will be having the address of the targeted node.

**Variables inside function:**

There will be some variables which will be used inside function:

1. **First Variable (int \*ptr):**First variable will be a pointer variable ‘ptr’ which will store the address of the particular node while searching or doing other operations with or with the help of it.
2. **Second Variable (int \*ptrsave):**Second Variable will be the pointer variable ‘ptrsave’ which will point to the parent of any particular node which is on operation.

**4 CASES:**

There are 4 cases in find function:

1. **Case 1: Empty Tree:**If tree is empty, then it will equate the loc and par pointers with NULL and the function will end.  
   loc = NULL  
   par = NULL
2. **Case 2: Root has the Targetted Value:**

If root has the value that needs to be searched, so loc will be set to address of root node and as it has no parent, so par will be set to NULL and function will stop.  
loc = root  
par = NULL

1. **Case 3: Tree has the Value anywhere except root:**If tree has the value somewhere else in the tree, then the loc variable will be set to the targeted node and the par variable will be set to parent of that targeted node and function will stop.  
   loc = targeted node  
   par = parentNode of targeted node
2. **Case 4: Tree does not has the value:**If tree does not has the targeted value, then the loc variable will be set to NULL and the par variable will be set to the last node which was checked while searching.  
   loc = NULL  
   par = last node checked

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Search Function:**

Find function will take 2 variables in parameters.

1. **First Variable (int data)**  
   One variable will be the value of targeted value which needs to be searched and it is ‘data’.
2. **Second Variable (int \*root)**  
   Second variable will be the pointer variable which will store the address of root of a tree, its name is ‘root’.

**Variables inside function:**

There will be some variables which will be used inside function:

1. **First Variable (int depth):**First variable will be a int variable ‘depth’ which will store the depth of the particular node while searching or doing other operations with or with the help of it.
2. **Second Variable (int \*temp):**Second Variable will be the pointer variable ‘temp’ which will point to the address of any particular node which is on operation.

**2 CASES:**

There are 2 cases in find function:

1. **Case 1: Targeted Value Found:**If value is found, it will print that data found at any depth and function will be stopped.
2. **Case 2: Targeted Value not found:**If value is not found, then it will print that value not found.