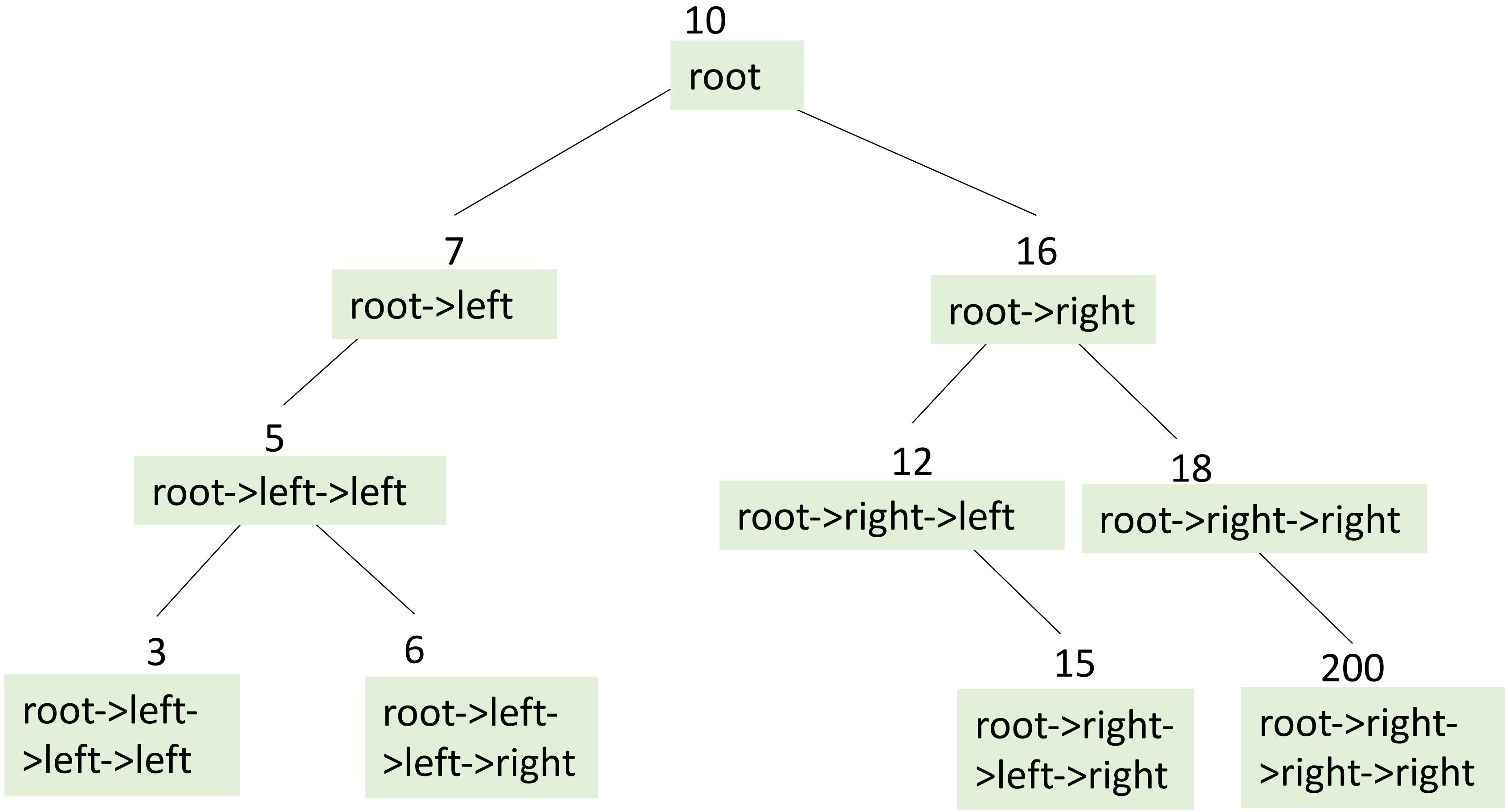
CSE 3010 – Data Structures & Algorithms

Lecture #39

Guided practice session on BST

Input dataset: 10, 7, 5, 16, 3, 6, 12, 15, 18, 200



10

root

7

root->left

16

root->right

5

root->left->left

12

root->right->left

18

root->right->right

3

root->left->left->left

6

root->left->left->right

15

root->right->left->right

200

root->right->right->right

Steps to understand the use of various functions of a BST

Open the `BST_application.c`

Declare the following variables:

```
BSTNODE *root, *node;

ITEM key;
```

Create a binary search tree and read data into the tree

```
root = createBSTree();
root = readData(root);
```

Check if the data is read properly into the tree by traversing in order

```
printf("\nPrinting the elements in ascending order\n");

inOrderTraversal(root);

printf("\n");
```

Steps to understand the use of various functions of a BST

Find if a key exists in the binary search tree

```
printf("Enter key to search in BST: ");
    scanf("%d",&key);

if (findNode(root,key))
   printf("%d found in BST!\n", key);
else
   printf("%d not found in BST\n!",key);
```

Get left or right child of a node

```
node = root;
printf("Left child of [%d] is [%d]\n",node->key,
getLeftChild(node)->key);

node = root->right;
printf("Right20 child of [%d] is [%d]\n",node->key,
getRightChild(node)->key);
```

Steps to understand the use of various functions of a BST

Checking size of tree

```
printf("Size of tree is %d\n\n",sizeTree(root));
```

Delete node from the binary search tree

```
deleteNode(root,6);
printf("Size of tree is %d\n\n",sizeTree(root));
inOrderTraversal(root);
deleteNode(root,6);
printf("Size of tree is %d\n\n",sizeTree(root));
inOrderTraversal(root);
```