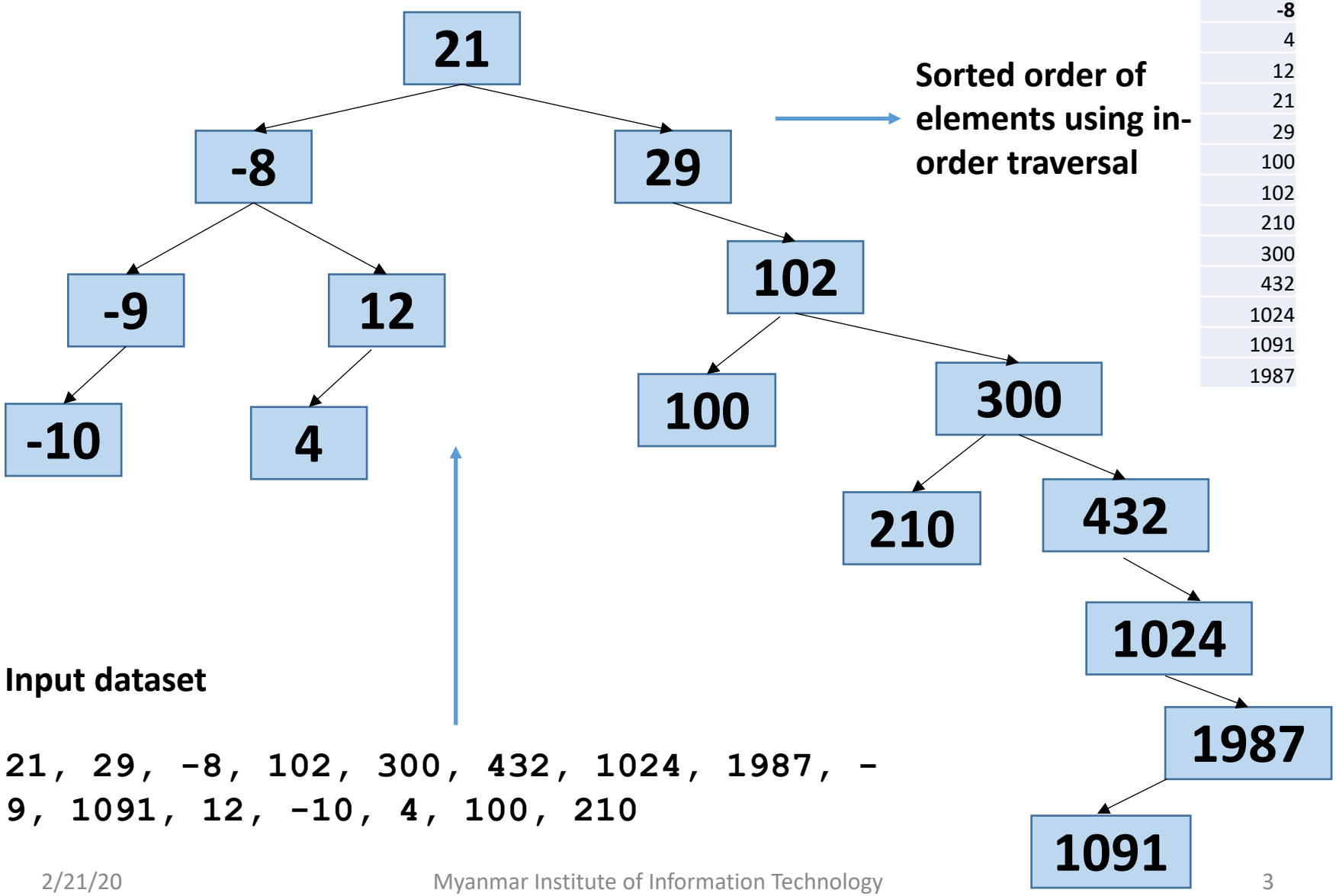
CSE 3010 – Data Structures & Algorithms

Lecture #33

What will be covered today

- Constructing a binary search tree
- C implementation of a binary search tree
  - Traversing a binary search tree
  - Creating a binary search tree node
  - Inserting into a binary search tree
- Assignment 3
  - On in-place sorting using doubly linked list

inOrderTraversal – Binary Search Tree



Traversing a binary search tree

```
// Traversal of tree using inorder method
void inOrderTraversal(BSTNODE *root) {
    if (root != NULL) { // Terminating condition
        inOrderTraversal(root->left);
        printf("\t%d\n", root->key);
        inOrderTraversal(root->right);
    }
}
```

Traversing a binary search tree

```
// Traversal of tree using postorder method
void postOrderTraversal(BSTNODE *root) {
    if (root != NULL) { // Terminating condition
        postOrderTraversal(root->left);
        postOrderTraversal(root->right);
        printf("\t%d\n", root->key);
    }
}
```

Traversing a binary search tree

```
// Traversal of tree using preorder method
void preOrderTraversal(BSTNODE *root) {
    if (root != NULL) { // Terminating condition
        printf("\t%d\n", root->key);
        preOrderTraversal(root->left);
        preOrderTraversal(root->right);
    }
}
```

Create empty BST and BST node

```
// Create the binary search tree
BSTNODE* createBSTree() {
    return NULL;
}

// Create a new tree node
BSTNODE* createNode(ITEM key) {

    BSTNODE *temp = (BSTNODE*) malloc(sizeof(BSTNODE));
    temp->key = key;
    temp->left = NULL;
    temp->right = NULL;
    return temp;
}
```

Insert a node in BST

```
// Insert a node in the tree
BSTNODE* insertItem(BSTNODE *root, ITEM key) {

    if (root == NULL)
        root = createNode(key);
    else {
        if (key < root->key)
            root->left = insertItem(root->left, key);
        else
            root->right = insertItem(root->right, key);
    }
    return root;
}
```

Will be the root of the subtree, if NULL item inserted else continue until a NULL on the subtree is found