Virtual University	**Assignment No. 02** **SEMESTER Fall 2025** **CS301P - Data Structures (Practical)**	**Total Marks: 20** **Due Date: 22 - Dec, 2025**

Instructions

Please read the following instructions carefully before solving & submitting assignment:

It should be clear that your assignment will not get any credit (zero marks) if:

- The assignment is submitted after the due date.
- The submitted code does NOT compile.
- The submitted assignment is other than .CPP file.
- The submitted assignment does NOT open or file is corrupted.
- The assignment is copied (from other student or ditto copy from handouts or internet).
- The BST is implemented using array and STL container of BST is used.

Uploading instructions

For clarity and simplicity, you are required to Upload/Submit **only ONE .cpp " Assignment 2 Template"** file after completing its code.

Note:

- Use ONLY Dev-C++ IDE.
- Only add the code where commented (// Write your code here) in given .cpp "Assignment 2 Template" file. Don't change any other code otherwise you will lose your marks.
- Mention your own student VU id instead of BC123456789 in "Assignment 2 Template" file as given in the output screenshot.
- Only implement these functions insertRec(), deleteRec(), searchRec(), findMaxNum().

Learning Objectives:

- Understand and implement a Binary Search Tree (BST) using linked nodes
- Practice recursive operations: insert, delete, search, and traversal
- Strengthen pointer manipulation and tree navigation concepts
- Display sorted data using inorder traversal
- Retrieve the maximum value from a BST efficiently

For any query about the assignment, contact at cs301P@vu.edu.pk

GOOD LUCK

Problem Statement **Marks 20**

You are working in the IT department of a school's digital record system, where you are required to develop a **Number Management System** using a **Binary Search Tree (BST)** implemented with linked nodes. This system stores and manages unique numbers by performing BST operations such as insertion, deletion, searching, inorder traversal, and finding the maximum value.

The system performs the following operations sequentially, as outlined in the provided "**Assignment 2 Template.cpp**" file:

1. Initialize and Display BST
    - Inserts the following predefined fixed numbers into the BST:
      50, 30, 70, 20, 40, 60, 80
    - Displays the BST using inorder traversal, which shows the numbers in sorted order.
2. Delete a Number
    - Removes the number 30 from the BST using the delete operation.
    - Displays the updated BST (using inorder traversal) after deletion.
3. Search for a Number
    - Searches for the number 70 inside the BST.
    - Displays "Found!" if the number exists, otherwise "Not Found!".
4. Find Maximum Value
    - Retrieves and displays the maximum value stored in the BST by traversing to the right-most node.

Your Task to Perform:

• Understand the Provided Code Template:

Open and carefully understand the "Assignment 2 Template.cpp" file provided within the assignment .zip file uploaded on VULMS. This template file includes the foundational class structure for implementing the Number Management System using a Binary Search Tree (BST) implemented using linked list.

• Implement insertRec() Function:

Write the code to insert a new number into the BST using recursion, ensuring the BST property (left < root < right) is maintained.

• Implement deleteRec() Function:

Write the code to delete a number from the BST using recursion.

Your code must correctly handle all three deletion cases:

1. Node with no child
2. Node with one child

3. Node with two children (replace with inorder successor)

• Implement searchRec() Function:

Write the code to search for a number in the BST using recursion and return the node if found.

• Implement findMaxNum() Function:

Write the code to find the maximum value in the BST by traversing to the right-most node.

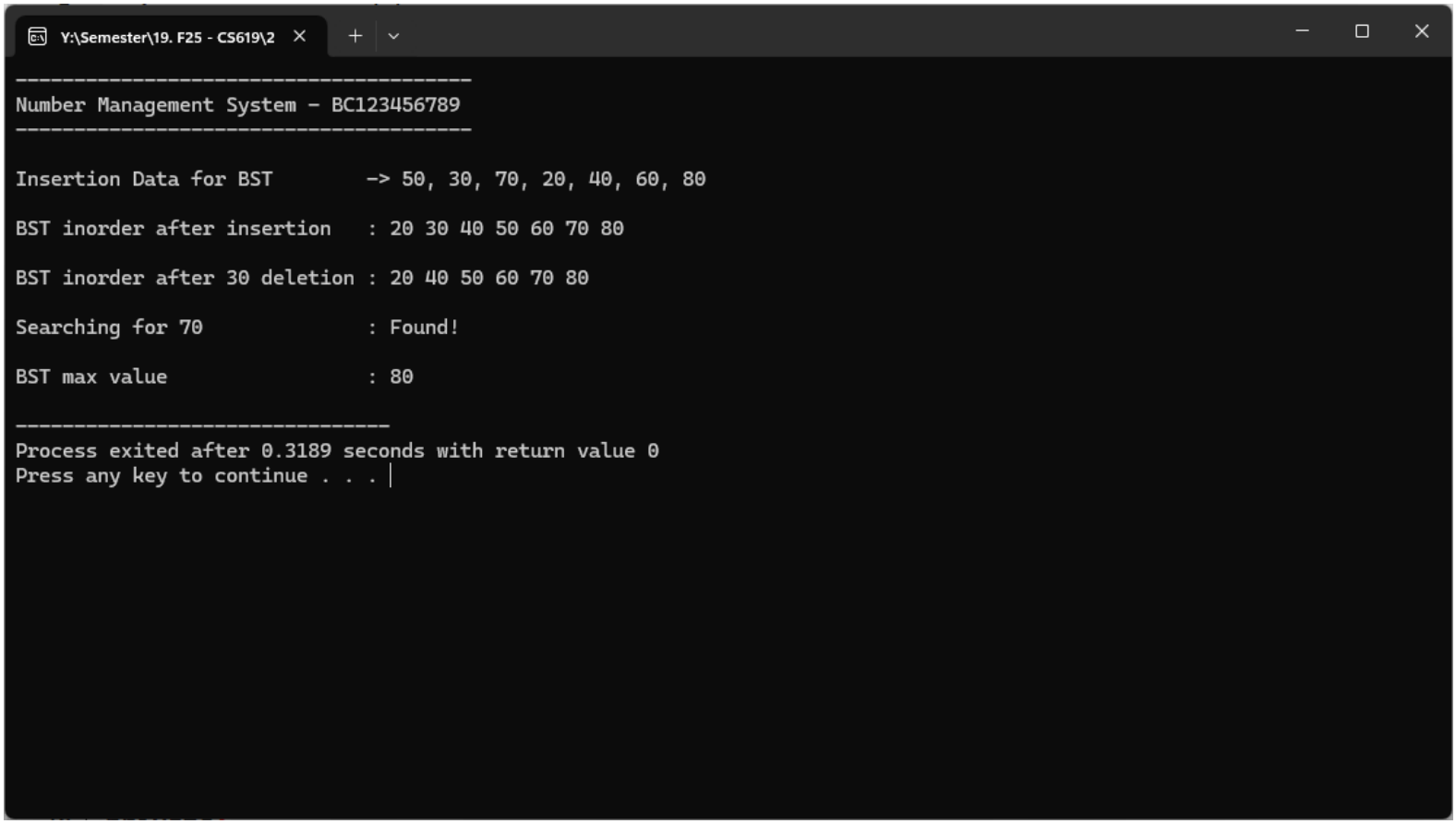
• In the main function:

- Replace the Sample VUID with Your Own:
  - In the code and output messages, replace the example VUID "BC123456789" with **your actual VUID** to personalize and validate your submission. Otherwise, your marks will be deducted.

```
// TODO 5: Write your VUID
string studentID = "BC123456789"; // Replace BC123456789 with your VUID
```

• Check the sample output provided in the assignment file to understand the expected program behaviour and formatting.

Sample Output Screenshot:



Lectures Covered: This assignment covers Lab 4 - 6.
Deadline: Your assignment must be uploaded/submitted on or before 22 Dec 2025.