Problems on Tree

1. Write a C program that creates a binary tree. Allow users to input nodes and build a binary tree structure.

2. Write a C program to perform an in-order traversal of a binary tree. Print the elements in sorted order.

3. Write a C program that extends the binary tree program to support the insertion of elements. This is in a way that maintains the binary search tree property.

4. Write a C program to calculate the height of a binary tree. Ensure the program handles empty trees gracefully.

5. Write a C program that implements a deletion function for a binary tree. Allow users to delete nodes while maintaining the binary search tree structure.

6. Write a C program to create a mirror image of a binary tree. Print both the original and mirrored trees.

7. Write a C program that extends the binary tree program to perform a level-order traversal. Print the nodes at each level from top to bottom.

8. Write a C program to build an expression tree from a postfix expression. Evaluate and display the results.

9. Write a C program to determine if a binary tree has a root-to-leaf path whose sum equals a given target sum.

10. Write a C program that implements an AVL tree in C. Include functions for insertion and deletion while maintaining the AVL tree's balance property.