``````````````````

**Que. 1:** Write C/C++ Code/ Algorithm for  Deleting a node from a Binary Search Tree (BST) with following  Situations:

| **S.No.** | **Condition** |
|---|---|
| A. | Tree is empty |
| B. | Key is smaller than the current node’s value |
| C. | Key is larger than the current node’s value |
| D. | Key is not found |
| E. | Node is a leaf (no children) |
| F. | Node has only a left child |
| G. | Node has only a right child |
| H. | Node has two children |

**Que. 2:**
``````````````````

Condition	Action
Tree is empty	**Return NULL.**
Key is smaller than the current node's value	**Search and delete in the left subtree.**
Key is larger than the current node's value	**Search and delete in the right subtree.**
Key is not found	**Leave the tree unchanged.**
Node is a leaf (no children)	**Delete the node and return NULL.**
Node has only a left child	**Delete the node and replace it with its left child.**
Node has only a right child	**Delete the node and replace it with its right child.**
Node has two children	**Copy its inorder successor's value, then delete the successor from the right subtree.** **Alternatively, use its inorder predecessor.**