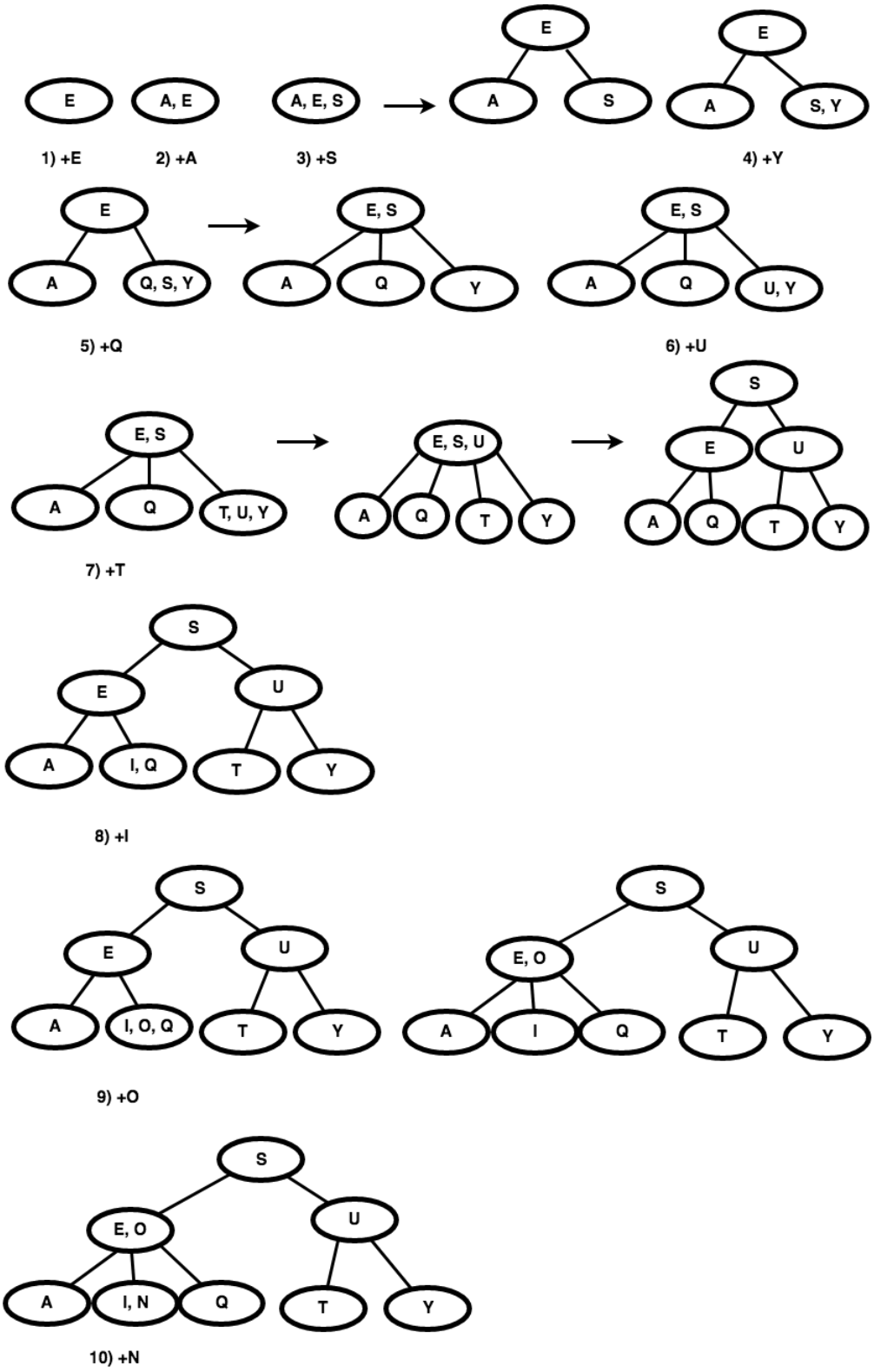
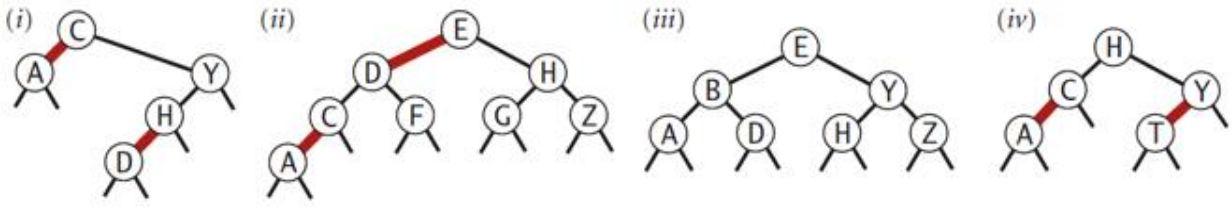
1. Draw the 2-3 tree that results when you insert the keys E A S Y Q U T I O N in that order into an initially empty tree.

Answer:



2. Which of the following are red-black BSTs? Convert each red-black BST to a 2-3 tree.



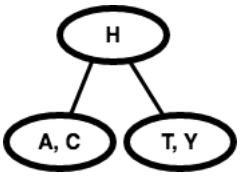
Answer:

(i) is not a red-black BST, because it does not have perfect black balance

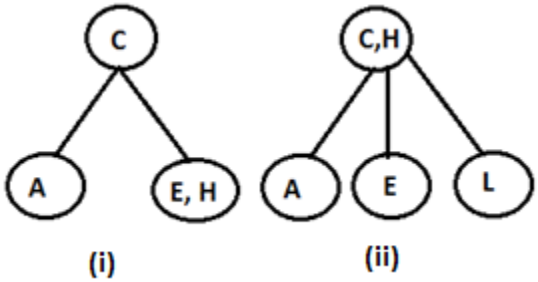
(ii) is not a red-black BST because two reasons. First, it does not follow BST key order, for instance F should not be inside E's left sub-tree. Second, it does not have perfect black balance.

(iii) is a red-black BST, and the converted 2-3 tree looks the same as the original one.

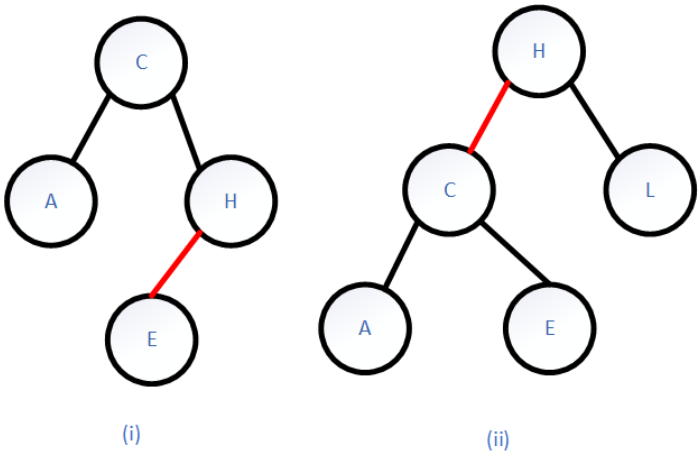
(iv) is a red-black BST, and the following is the converted 2-3 tree



3. Convert each of the following 2-3 tree to a red-black BST.



Answer:



4. If you insert keys in increasing order into a red-black BST, the tree height increases after each insertion.

Answer: That statement is false. If the tree increases height after each insertion, then each node can have a most one child, which cannot have perfect black balance. Thus, that cannot be the case for red-black BST.

5. Draw the Red-Black BST that results from inserting the keys E A S Y Q U T I O N in that order into an initially empty tree.

Answer:

See the attached RedBlackBSTInsertionExample_Exercise.jpg.