

Algorithm for Insertions into a Patricia Tree

- 1) First, check to see if your tree is empty. If it is then find the highest order bit in which your new value has a one. This is the highest bit in which it differs from the “dummy” all zero node. Allocate a node that root now points to, insert your value and the bit test # you just found. Make the right child (child for 1's) point to the new root and the left child (child for 0's) point to the dummy all zero header node.
- 2) Otherwise, your tree is not empty. Search in the existing tree for the New Item you are inserting, keeping track of the bit tests as you search.
- 3) Once the most recent bit test is equal to or higher numerically than the previous bit test, stop your search.
- 4) Compare the Old Item you finished your search with to the New Item you are inserting in ALL bits. Find the **highest order bit, n** – numerically – in which the New Item differs from the Old Item
- 5) Now back up through the search path of Old Items until you identify where in the path this bit test would occur – Cases:
 - a) The bit test n is **higher** (greater than) any other bit test in the existing tree: NEW ROOT CASE
 - b) The bit test is **lower** (less than) any bit test on the search path: Unwrap the last test on this path and attach your New Item with this lower untested bit labeled on the item. Assign left child (0) and right child (1) values pointers accordingly.
 - c) The bit test is **“in-between”** existing bit tests on the search path: Retrace backward along the search path to where this bit test would occur. Insert the New Item “between” the existing values which we will call “old parent” and “old child”. Attach the appropriate bit test to your New Item. If your New Item has a 1 in this bit, put a self-loop from the right child to this New Item and attach the “old child” as a left child, since it has a 0 in this bit. Similarly, if your New Item has a 0 in this bit, put a self-loop from the left child to this New Item and attach the “old child” as a right child, since it has a 1 in this bit.