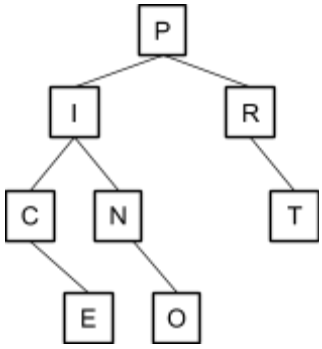
Binary Search & Binary Trees Solution

1. Starting from an empty binary search tree, create the binary search tree with the letters:

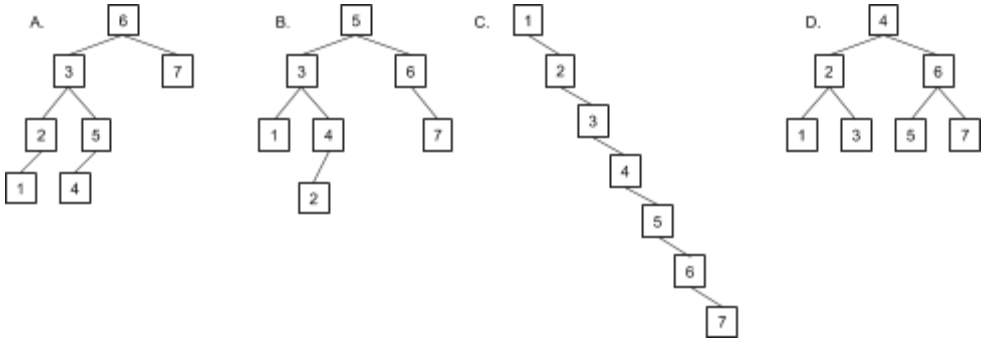
   P, R, I, N, C, E, T, O, N



   a. What keys are examined when we search for E? **P I C E**

   b. What keys are examined when we search for Q? **P R**

2. Which of the following is not a valid binary search tree? Of the valid ones, which is fastest to search?



Answer: B is invalid — 2 cannot be found when we search for it. For performance, it depends on how often each key occurs. If all occur with the same probability, the D is the best. C requires sequential search, but, yes, valid BST.

3. Suppose we have integer values between 1 and 1000 in a BST and search for 363. Which of the following cannot be the sequence of keys examined?

   a. 2, 252, 401, 398, 330, 363

   b. 399, 387, 219, 266, 382, 381, 278, 363

   c. 3, 923, 220, 911, 244, 898, 258, 362, 363

   d. 4, 924, 278, 347, 621, 299, 392, 358, 363

   - 299 cannot appear after 621 since that would place it to the right of 347.

   e. 5, 925, 202, 910, 245, 363