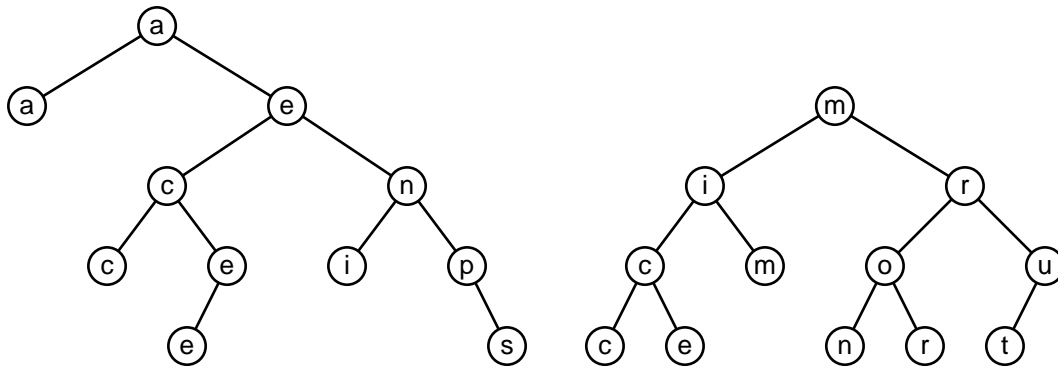


“A Balancing Act”

Problem: A tempting approach to maintaining a balanced binary search tree is to maintain two binary search trees and to insert each new key into the tree that will be more balanced.

More specifically, do the following: The first key is the root of the left tree. The second key is the root of the right tree. To add a new key, insert it into the tree where it would have a smaller depth. If the depths of both trees are the same, then add it to the first tree.

For example, considering the following two trees, built from the string `american computer science`:



The input will be 5 strings (ignore everything but the letters A through Z and a through z; uppercase and lowercase are the same). For each input string, build the two trees with each string as described above. Print contents of the first tree in preorder (root, then the left child, then the right child).

Sample Input:

Line #1: AMERICAN
 Line #2: AMERICAN COMPUTER SCIENCE
 Line #3: INTERMEDIATE
 Line #4: DIVISION
 Line #5: INTERMEDIATE DIVISION

Sample Output:

Output #1: A A E C N
 Output #2: A A E C C E E N I P S
 Output #3: I E E A I T T
 Output #4: D V I N
 Output #5: I E E A I I I T T V

“A Balancing Act”

Test Input:

Line #1: Time flies like an arrow. Fruit flies like a banana.

Line #2: I must say that I find television very educational.

Line #3: The minute somebody turns it on, I go to the library and read a book.

Line #4: One morning I shot an elephant in my pajamas.

Line #5: How he got into my pajamas I'll never know!

Test Output:

Output #1: T M F E A A A E F I I I L S R O S T W U

Output #2: I A A A A I D D C E E I U T T S R U Y V Y

Output #3: T E E D A A D E I I I T O O O R R R T U U Y Y

Output #4: O E A A A A E E N G H I R P P S S T

Output #5: H H G A E W O N M M L N N O P R Y