Questions on Binary Trees and Binary Search Trees

1) Which of the following statements about binary trees is correct?

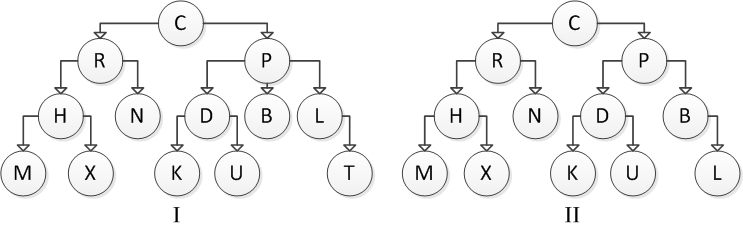
a) Each node in a binary tree has at least two child nodes.

**b) Each node in a binary tree has at most two child nodes**.

c) The number of child nodes for each node in a binary tree is any power of two.

d) If divided down the middle from top to bottom, a binary tree must be symmetrical.

2) Consider the following tree diagrams:



Which of the above are binary trees?

a) I

**b) II**

c) I and II

d) Neither I nor II

3) The following values are inserted into a binary search tree. Show the result after each insertion.

Ann, Eve, Rob, Joe, Tom, Bob, Jim

**Anna**

**/ \**

**Rob Eve**

**/**

**Joe**

**/ \**

**Bob Tom**

**/**

**JIm**

What is the height of the resulting tree? Is it full, complete or balanced? **3**

4) Same as Q3 but for the following numbers:

100, 76, 110, 105, 104, 130, 84.

**100**

**/ \**

**76 110**

**/ / \**

**84 105 130**

**/**

**104**

5) A balanced binary tree with 520 nodes has a height of approximately \_\_\_\_.

a) 9

**b) 10**

c) 12

d) 13

6) A binary tree with 260 nodes has a height of approximately \_\_\_\_.

a) 8

b) 10

c) 12

**d) 13**

7) A binary tree of height *h* can have up to \_\_\_\_ nodes.

a) *2h -* 1

b) *2h +* 1

**c) *2h -* 1**

d) *2h +* 1

8) The height *h* of a completely filled binary tree with *n* nodes is \_\_\_\_.

**a) *h* = log2(*n*) - 1**

b) *h* = log2(*n)* + 1

c) *h* = log2(*n* – 1)

d) *h* = log2(*n* + 1)

9) A completely filled binary tree with a height of 3 has \_\_\_\_ nodes.

a) 6

**b) 7**

c) 8

d) 12

10) A completely filled binary tree with a height of 4 has \_\_\_\_ nodes.

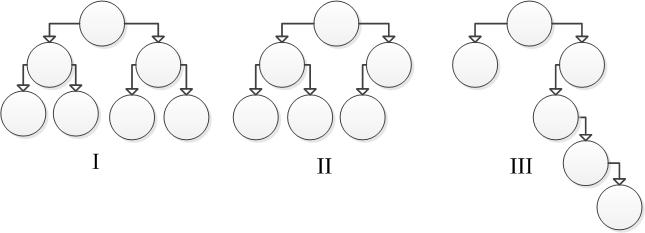
a) 8

b) 12

**c) 15**

d) 16

11) Consider the following tree diagrams:



Which of these trees is considered to be balanced?

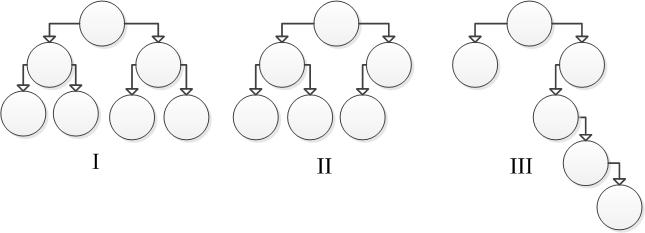
a) I

**b) I and II**

c) II and III

d) I and III

12) Consider the following tree diagrams:



Which of these trees is considered to be unbalanced?

a) I

b) II

**c) III**

d) II and III

13) If the child references of a binary tree node are both null, the node is \_\_\_\_.

a) a root node

**b) a leaf node**

c) a parent node

d) an interior node

14) If both of the child references of a binary tree node are non-null, it follows that the node must be \_\_\_\_.

a) a root node

b) a leaf node

c) a child node

**d) an interior node**

15) In a binary search tree, where the root node data value = 45, what do we know about the data values of all the descendants in the left subtree of the root?

a) the root’s left child value < 45, but the right child of the root’s left child value is > 45

b) some values will be < 45, but there may be a few values > 45

c) approximately half the values are < 45, the other half are > 45

**d) all will be < 45**

16) In a binary search tree, where the root node data value = 45, what do we know about the values of all the descendants in the right subtree of the root?

a) the root’s right child value > 45, but the left child of the root’s right child key is < 45

b) some values will be > 45, but there may be a few values < 45

c) approximately half the values are < 45, the other half are > 45

**d) all will be > 45**

17) A binary search tree is made up of a collection of nodes organized with smaller data values on the left and greater values on the right, relative to any node. Which of the following Node references must be instance variables of any implementation of a BinarySearchTree class?

I root

II left

III right

a) I

b) II and III

c) I and II

**d) I, II and III**

18) A binary search tree is made up of a collection of nodes organized with smaller data values on the left and greater values on the right relative to any node. Which of the following Node references must be instance variables of any implementation of a Node class?

I root

II left

III right

a) I

**b) II and III**

c) I and II

d) I, II and III

19) The nodes in our binary search tree implement the Comparable interface. Which tree operations benefit from this design decision?

I add

II search

III delete

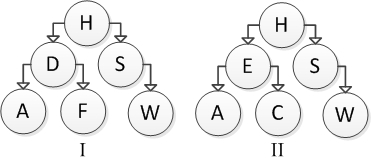
a) I

b) II

c) I and III

**d) I, II and III**

20) Consider the following tree diagrams:



Which are binary search trees?

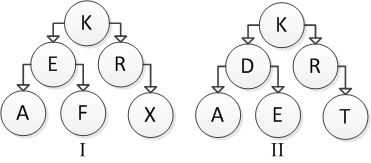
a) I

b) II

**c) I and II**

d) Neither I nor II

21) Consider the following tree diagrams:



Which of the above are binary search trees?

**a) I**

b) II

c) I and II

d) Neither I nor II

22) Consider the following binary search tree diagram:



Which nodes will be visited in order to insert the letter B into this tree?

a) H

b) H and D

c) H, D, and F

**d) H, D, and A**

23) Consider the following binary search tree diagram:



Which nodes will be visited in order to insert the letter E into this tree?

a) H

b) H and D

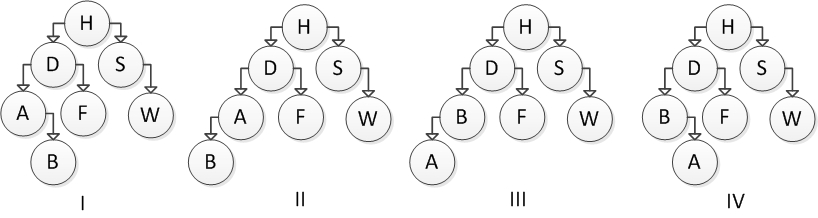
**c) H, D, and F**

d) H, D, and A

24) Consider the following binary search tree diagram:



Which of the following trees represents the correct result after inserting element B?



a) I

b) II

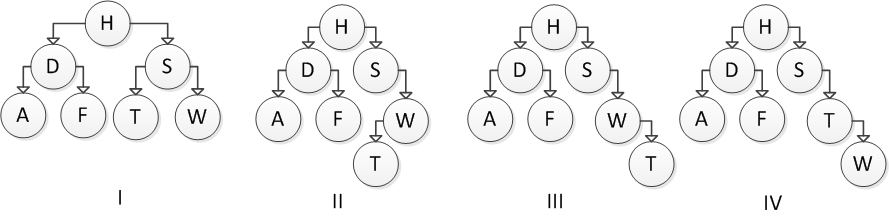
**c) III**

d) IV

25) Consider the following binary search tree diagram:



Which of the following trees represents the correct result after inserting element T?



a) I

b) II

c) III

**d) IV**

26) What does the left node reference of a newly inserted binary search tree node get set to?

a) depends where the node is inserted

b) it gets set to the left child of the new node, if one exists

**c) always null**

d) it gets set to the left child of the root, if it exists

27) Which of the following may occur as a result of an add operation, on a non-empty binary search tree?

I a new root is created

II the new node becomes the left child of the root

III the new node has a right child upon insertion

a) I

**b) II**

c) III

d) II and III

28) Which of the following sequences of insertions will result in a balanced tree?

I 12 , 7, 25, 6, 9, 13, 44

II 12 , 7, 25, 44, 13, 6, 9

III 12, 25, 44, 13, 6, 9, 7

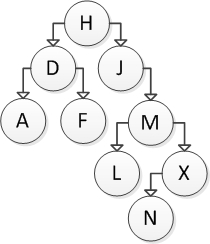
a) I

b) II

**c) I and II**

d) I and III

29) Consider the following binary search tree diagram:



If node J is to be removed, which node should be copied into its location?

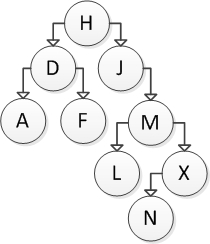
**a) M**

b) L

c) X

d) N

30) Consider the following binary search tree diagram:



If node M is to be removed, which action should be taken?

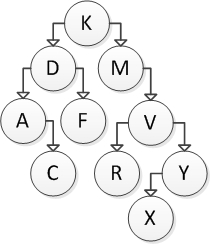
a) Replace M with the smallest value in its left subtree.

**b) Replace M with the smallest value in its right subtree.**

c) Replace M with the largest value in its left subtree.

d) Replace M with the largest value in its right subtree.

31) Consider the following binary search tree diagram:



If node F is to be removed, which action should be taken?

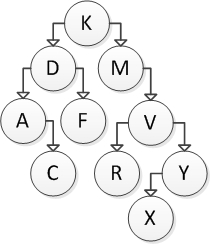
a) Move C into the right subtree of D.

b) Move C into the left subtree of A.

c) Replace F with D’s value and replace D with C’s value.

**d) Modify D to have a null right reference.**

32) Consider the following binary search tree diagram:



If node Y is to be removed, which action should be taken?

a) Modify the V’s left reference to point to X.

**b) Modify the V’s right reference to point to X.**

c) Swap the values in V and X, and modify X’s right reference to point to V.

d) Modify V to have a null right pointer.

33) Which of the following statements about a binary search tree is correct?

a) Adding elements that are already sorted will result in a balanced binary search tree.

b) Nodes must be moved when a node is removed from the middle of a subtree.

**c) The speed of inserting or removing a node is dependent on the shape of the tree.**

d) The speed of inserting or removing a node is dependent on the number of subtrees.

34) Locating an element in a balanced binary search tree takes \_\_\_\_ time.

a) *O*(*n*)

**b) *O*(log(*n)*)**

c) *O*(1)

d) *O*(*n*2)

35) Adding an element to a balanced binary search tree takes \_\_\_\_ time.

a) *O*(*n*)

b) *O*(log (*n*))

c) *O*(1)

d) *O*(*n*2)

36) Locating an element in an unbalanced binary search tree takes \_\_\_\_ time.

**a) *O*(*n*)**

b) *O*(log (*n)*)

c) *O*(1)

d) *O*(*n*2)

37) Adding an element to an unbalanced binary search tree takes \_\_\_\_ time.

**a) *O*(*n*)**

b) *O*(log (*n*))

c) *O*(1)

d) *O*(*n*2)

38) You wish to traverse a binary search tree in sorted order using inorder traversal. Arrange the following actions in the correct order to accomplish this.

I Print the right subtree recursively

II Print the root

III Print the left subtree recursively

a) I, II, III

**b) III, II, I**

c) II, III, I

d) III, I, II

39) You wish to traverse a binary search tree in sorted order. Which of the following schemes will accomplish this?

I inorder traversal

II preorder traversal

III postorder traversal

**a) I**

b) II

c) III

d) II and III

40) You wish to traverse a binary search tree in sorted order using preorder traversal. Arrange the following actions in the correct order to accomplish this.

I Print the right subtree recursively

II Print the root

III Print the left subtree recursively

a) I, II, III

b) III, II, I

**c) II, III, I**

d) III, I, II

41) You wish to traverse a binary search tree using postorder traversal. Arrange the following actions in the correct order to accomplish this.

I Print the right subtree recursively

II Print the root

III Print the left subtree recursively

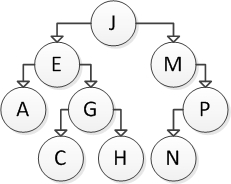
a) I, II, III

b) III, II, I

c) II, III, I

**d) III, I, II**

42) Consider the following binary search tree:



Which of the following sequences correctly represents preorder traversal of this tree?

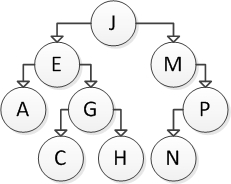
a) J, E, M, A, G, P, C, H, N

b) A, C, H, G, E, N, P, M, J

**c) J, E, A, G, C, H, M, P, N**

d) A, C, E, G, H, J, M, N, P

43) Consider the following binary search tree:



Which of the following sequences correctly represents postorder traversal of this tree?

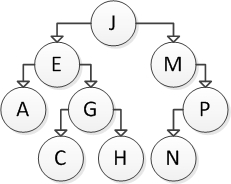
a) J, E, M, A, G, P, C, H, N

b) A, C, H, G, E, N, P, M, J

**c) J, E, A, G, C, H, M, P, N**

d) A, C, E, G, H, J, M, N, P

44) Consider the following binary search tree:



Which of the following sequences correctly represents breadth-first traversal of this tree?

a) J, E, M, A, G, P, C, H, N

b) A, C, H, G, E, N, P, M, J

c) J, E, A, G, C, H, M, P, N

**d) A, C, E, G, H, J, M, N, P**

45) What are the differences between preorder, postorder, and inorder traversals?

a) The order in which we visit the left and right subtrees

b) Preorder only visits the left subtree

c) Postorder only visits the right subtree

**d) The order of the root visit**

46) If the postorder traversal visits the nodes of a binary tree storing character values in the order of U, G, T, R, A, I, what is the value of the root of the tree?

**a) I**

b) U

c) T

d) cannot be determined

47) If the postorder traversal visits the nodes of a binary tree storing character values in the order of

U, G, T, R, A, I, what is the visit order for an inorder traversal of the same binary tree?

**a) I, G, U, A, T, R**

b) R, G, U, I, T, A

c) G, U, I, T, A, R

d) cannot be determined