1) The number of ways in which the numbers 1, 2, 3, 4, 5, 6, 7 can be inserted in an empty binary search tree, such that the resulting tree has height 6, is \_\_\_\_\_\_\_\_\_\_\_\_\_

Note: The height of a tree with a single node is 0.

(A) 2  
(B) 4  
(C) 64  
(D) 32

Answer:(C)  
  
Explanation: To get height 6, we need to put either 1 or 7 at root.

So count can be written as T(n) = 2\*T(n-1) with T(1) = 1

7

/

[1..6]

1

\

[2..7]

Therefore count is 26 = 64

2) Suppose that we have numbers between 1 and 100 in a binary search tree and want to search for the number 55. Which of the following sequences CANNOT be the sequence of nodes examined?

(A) {10, 75, 64, 43, 60, 57, 55}

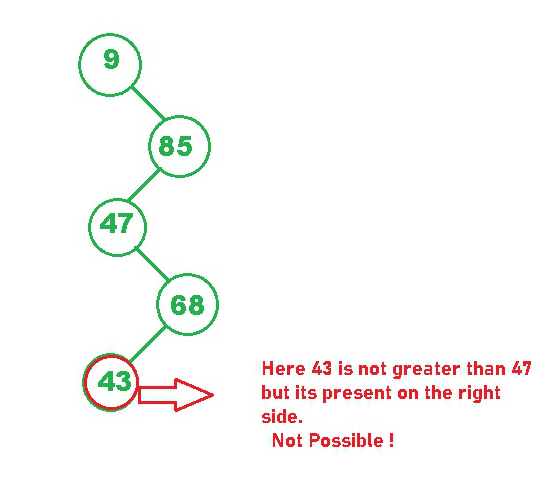
(B) {90, 12, 68, 34, 62, 45, 55}

(C) {9, 85, 47, 68, 43, 57, 55}

(D) {79, 14, 72, 56, 16, 53, 55}

Answer: (C)

Explanation: In BST, on right child of parent should be greater than parent and left child should be smaller than the parent, but in C after 47, 68 goes on the right side because it greater then parent, now everything below this point should be greater then 47 but 43 appears that does not satisfy the BST property.



3) Given an array of element 5, 7, 9, 1, 3, 10, 8, 4. Which of the following is the correct sequences of elements after inserting all the elements in a min-heap?

A) 1,3,4,5,7,8,9,10

B) 1,4,3,9,8,5,7,10

C) 1,3,4,5,8,7,9,10

D) 1,3,7,4,8,5,9,10

Answer: A

Explanation: Building a min-heap the result will a sorted array so the 1, 3, 4, 5, 7, 8, 9, 10 is correct. If we change the implementation strategy 1, 4, 3, 8, 9, 5, 7, 10 is also correct. (First filling the right child rather than left child first).

Explanation: As 35 is greater than 3, so there is a violation and the node will swap at that position. So leaf nodes with value s are 2 , 7 and 3.

4)Consider the Quick sort algorithm which sorts elements in ascending order using the first element as pivot. Then which of the following input sequence will require a maximum number of comparisons when this algorithm is applied on it?

A) 22 25 56 67 89

B) 52 25 76 67 89

C) 22 25 76 67 50

D) 52 25 89 67 76

Answer: A

Explanation: If the input sequence is already sorted then worst case behaviour occurs for the Quick sort algorithm which use the first element as pivot. Therefore, the input sequence given in 22 25 56 67 89 will require a maximum number of comparisons.

5)Consider two strings A = “qpqrr” and B = “pqprqrp”. Let x be the length of the longest common subsequence (not necessarily contiguous) between A and B and let y be the number of such longest common subsequences between A and B. Then x + 10y = \_\_\_.

A.24

B.34

C.44

D.23

Answer: B

The longest length is 4. There are 3 LCS of length 4 “qprr”, “pqrr” and “qpqr”.

6) Given:

1. public class Electronic implements Device

{ public void doIt() { } }

2.

3. abstract class Phone1 extends Electronic { }

4.

5. abstract class Phone2 extends Electronic

{ public void doIt(int x) { } }

6.

7. class Phone3 extends Electronic implements Device

{ public void doStuff() { } }

8.

9. interface Device { public void doIt(); }

What is the result? (Choose all that apply.)

A. Compilation succeeds

B. Compilation fails with an error on line 1

C. Compilation fails with an error on line 3

D. Compilation fails with an error on line 5

E. Compilation fails with an error on line 7

F. Compilation fails with an error on line 9

Answer:

A is correct; all of these are legal declarations.

7) Given:

public abstract interface Frobnicate { public void twiddle(String s); }

Which is a correct class? (Choose all that apply.)

A. public abstract class Frob implements Frobnicate {

public abstract void twiddle(String s) { }

}

B. public abstract class Frob implements Frobnicate { }

C. public class Frob extends Frobnicate {

public void twiddle(Integer i) { }

}

D. public class Frob implements Frobnicate {

public void twiddle(Integer i) { }

}

E. public class Frob implements Frobnicate {

public void twiddle(String i) { }

public void twiddle(Integer s) { }

}

Answer:

B is correct, an abstract class need not implement any or all of an interface’s methods.

E is correct, the class implements the interface method and additionally overloads the

twiddle() method.

A is incorrect because abstract methods have no body.

C is incorrect because classes implement interfaces they don’t extend them.

D is incorrect because overloading a method is not implementing it.

8) Given:

class Top {

public Top(String s) { System.out.print("B"); }

}

public class Bottom2 extends Top {

public Bottom2(String s) {

System.out.print("D"); }

public static void main(String [] args) {

new Bottom2("C");

System.out.println(" ");

} }

What is the result?

A. BD

B. DB

C. BDC

D. DBC

E. Compilation fails

Answer:

E is correct.

The implied super() call in Bottom2’s constructor cannot be satisfied because there isn’t a no-arg constructor in Top.

A default, no-arg constructor is generated by the compiler only if the class has no constructor defined explicitly.

9) You want to retrieve all employees, whether or not they have matching departments in the departments table.

Which query would you use?

A) SELECT last\_name, department\_name

FROM employees e LEFT OUTER

JOIN departments d ON (e.department\_id = d.department\_id);

B) SELECT last\_name, department\_name

FROM employees e RIGHT OUTER

JOIN departments d ON (e.department\_id = d.department\_id);

C) SELECT last\_name, department\_name

FROM employees e FULL OUTER

JOIN departments d ON (e.department\_id = d.department\_id);

D) None of these

Ans- A

10) What is the following function fun(int x, int y) calculate in C?

int fun(int x, int y)

{

if (y == 0) return 0;

return (x + fun(x, y-1));

}

(A) x + y

(B) x + x\*y

(C) x\*y

(D) xy

Answer: (C)

Explanation: The function adds x to itself y times which is x\*y.