Q.1	What are splay trees?	
	A	self adjusting binary search trees
	B	self adjusting binary trees
	C	a tree with strings
	D	a tree with probability distributions
Q.2	Which of the following property of splay tree is correct?	
	A	it holds probability usage of the respective sub trees
	B	any sequence of j operations starting from an empty tree with h nodes atmost, takes O(jlogh) time complexity
	C	sequence of operations with h nodes can take O(logh) time complexity
	D	splay trees are unstable trees
Q.3	Why to prefer splay trees?	
	A	easier to program
	B	space efficiency
	C	easier to program and faster access to recently accessed items
	D	quick searching
Q.4	Is it true that splay trees have O(logn) amortized complexity?	
	A	TRUE
	B	FALSE
Q.5	What is a splay operation?	
	A	moving parent node to down of child

	B	moving a node to root
	C	moving root to leaf
	D	removing leaf node
Q.6	Which of the following options is an application of splay trees?	
	A	cache Implementation
	B	Networks
	C	send values
	D	receive values
Q.7	When we have red-black trees and AVL trees that can perform most of operations in logarithmic times, then what is the need for splay trees?	
	A	no there is no special usage
	B	In real time it is estimated that 80% access is only to 20% data, hence most used ones must be easily available
	C	redblack and avl are not upto mark
	D	they are just another type of self balancing binary search trees
Q.8	After the insertion operation, is the resultant tree a splay tee?	
	A	TRUE
	B	FALSE
Q.9	What output does the below pseudo code produces?	
	Tree....node function(Tree....node x) { Tree....node y = x.left; x.left = y.right; y.right = x; return y; }	

	A	right rotation of subtree
	B	left rotation of subtree
	C	zig-zag operation
	D	zig-zig operation
Q.10	What is the disadvantage of using splay trees?	
	A	height of a splay tree can be linear when accessing elements in non decreasing order.
	B	splay operations are difficult
	C	no significant disadvantage
	D	splay tree performs unnecessary splay when a node is only being read
Q.11	What is the special property of red-black trees and what root should always be?	
	A	a color which is either red or black and root should always be black color only
	B	height of the tree
	C	pointer to next node
	D	a color which is either green or black
Q.12	Why do we impose restrictions like	
	. root property is black	
	. every leaf is black	
	. children of red node are black	
	. all leaves have same black	
	A	to get logarithm time complexity
	B	to get linear time complexity
	C	to get exponential time complexity

	D	to get constant time complexity
Q.13	Cosider the below formations of red-black tree.	
	All the above formations are incorrect for it to be a redblack tree. then what may be the correct order?	
	A	50-black root, 18-red left subtree, 100-red right subtree
	B	50-red root, 18-red left subtree, 100-red right subtree
	C	50-black root, 18-black left subtree, 100-red right subtree
	D	50-black root, 18-red left subtree, 100-black right subtree
Q.14	What are the operations that could be performed in O(logn) time complexity by red-black tree?	
	A	insertion, deletion, finding predecessor, successor
	B	only insertion
	C	only finding predecessor, successor
	D	for sorting
Q.15	Which of the following is an application of Red-black trees and why?	
	A	used to store strings efficiently
	B	used to store integers efficiently
	C	can be used in process schedulers, maps, sets
	D	for efficient sorting
Q.16	When it would be optimal to prefer Red-black trees over AVL trees?	
	A	when there are more insertions or deletions
	B	when more search is needed
	C	when tree must be balanced

	D	when log(nodes) time complexity is needed
Q.17	Why Red-black trees are preferred over hash tables though hash tables have constant time complexity?	
	A	no they are not preferred
	B	because of resizing issues of hash table and better ordering in redblack trees
	C	because they can be implemented using trees
	D	because they are balanced
Q.18	How can you save memory when storing color information in Red-Black tree?	
	A	using least significant bit of one of the pointers in the node for color information
	B	using another array with colors of each node
	C	storing color information in the node structure
	D	using negative and positive numbering
Q.19	When to choose Red-Black tree, AVL tree and B-trees?	
	A	many inserts, many searches and when managing more items respectively
	B	many searches, when managing more items respectively and many inserts respectively
	C	sorting, sorting and retrieval respectively
	D	retrieval, sorting and retrieval respectively
Q.20	What is the below pseudo code trying to do, where pt is a node pointer and root pointer	

```
redblack(Node root, Node pt) :
if (root == NULL)
return pt
if (pt.data < root.data)
{ root.left = redblack(root.left, pt); root.left.parent = root }
```

	else if (pt.data > root.data) { root.right = redblackt(root.right, pt) root.right.parent = root } return root	
	A	insert a new node
	B	delete a node
	C	search a node
	D	count the number of nodes
Q.21	In a B+ tree, both the internal nodes and the leaves have keys.	
	A	TRUE
	B	FALSE
	CORRECT	
Q.22	Which of the following is true?	
	A	B + tree allows only the rapid random access
	B	B + tree allows only the rapid sequential access
	C	B + tree allows rapid random access as well as rapid sequential access
	D	B + tree allows rapid random access and slower sequential access
Q.23	A B+ tree can contain a maximum of 7 pointers in a node. What is the minimum number of keys in leaves?	
	A	6
	B	3
	C	4
	D	7
Q.24	Which of the following is false?	
	A	A B+ -tree grows downwards
	B	A B+ -tree is balanced
	C	In a B+ -tree, the sibling pointers allow sequential searching

	D	B+ -tree is shallower than B-tree
Q.25	Statement 1: When a node is split during insertion, the middle key is promoted to the parent as well as retained in right half-node.	
	Statement 2: When a key is deleted from the leaf, it is also deleted from the non-leaf nodes of the tree.	
	A	Statement 1 is true but statement 2 is false
	B	Statement 2 is true but statement 1 is false
	C	Both the statements are true
	D	Both the statements are false
Q.26	Efficiency of finding the next record in B+ tree is	
	A	O(n)
	B	O(log n)
	C	O(nlog n)
	D	O(1)
Q.27	What is the maximum number of keys that a B+ -tree of order 3 and of height 3 have?	
	A	3
	B	80
	C	27
	D	26
Q.28	Which of the following is false?	
	A	Compared to B-tree, B+ -tree has larger fanout
	B	Deletion in B-tree is more complicated than in B+ -tree
	C	B+ -tree has greater depth than corresponding B-tree
	D	Both B-tree and B+ -tree have same search and insertion efficiencies

Q.29	Which one of the following data structures are preferred in database-system implementation?	
	A	AVL tree
	B	B-tree
	C	B+ -tree
	D	Splay tree
Q.30	What will be the height of a balanced full binary tree with 8 leaves?	
	A	8
	B	5
	C	6
	D	4
Q.31	The balance factor of a node in a binary tree is defined as	
	A	addition of heights of left and right subtrees
	B	height of right subtree minus height of left subtree
	C	height of left subtree minus height of right subtree
	D	height of right subtree minus one
Q.32	Figure below is a balanced binary tree. If a node inserted as child of the node R, how many nodes will become unbalanced?	
	A	2
	B	1
	C	3
	D	0
Q.33	A binary tree is balanced if the difference between left and right subtree of every node is not more than	
	A	1
	B	3

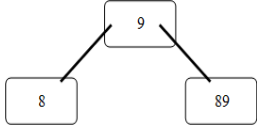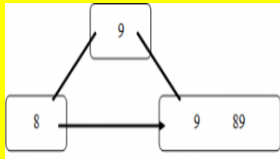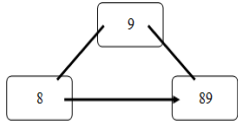|	C	2
	D	0
Q.34	Which of the following tree data structures is not a balanced binary tree?	
	A	AVL tree
	B	Red-black tree
	C	Splay tree
	D	B-tree
Q.35	Balanced binary tree with n items allows the lookup of an item in ..... worst-case time.	
	A	O(log n)
	B	O(nlog 2)
	C	O(n)
	D	O(1)
Q.36	Which of the following data structures can be efficiently implemented using height balanced binary search tree?	
	A	sets
	B	priority queue
	C	heap
	D	both sets and priority queue
Q.37	Two balanced binary trees are given with m and n elements respectively. They can be merged into a balanced binary search tree in ..... time.	
	A	O(m+n)
	B	O(mn)
	C	O(m)

	D	O(mlog n)
Q.38	Which of the following is an advantage of balanced binary search tree, like AVL tree, compared to binary heap?	
	A	insertion takes less time
	B	deletion takes less time
	C	searching takes less time
	D	construction of the tree takes less time than binary heap
Q.39	AVL trees are more balanced than Red-black trees.	
	A	TRUE
	B	FALSE
	Top of Form	
Q.40	Which of the following is true?	
	A	B + tree allows only the rapid random access
	B	B + tree allows only the rapid sequential access
	C	B + tree allows rapid random access as well as rapid sequential access
	D	B + tree allows rapid random access and slower sequential access
Q.41	A B+ tree can contain a maximum of 7 pointers in a node. What is the minimum number of keys in leaves?	
	A	6
	B	3
	C	4
	D	7
Q.42	Which of the following is false?	
	A	A B+ -tree grows downwards
	B	A B+ -tree is balanced

	C	In a B+ -tree, the sibling pointers allow sequential searching
	D	B+ -tree is shallower than B-tree
Q.43	Efficiency of finding the next record in B+ tree is _____	
	A	O(n)
	B	O(log n)
	C	O(nlog n)
	D	O(1)
Q.44	What is the maximum number of keys that a B+ -tree of order 3 and of height 3 have?	
	A	3
	B	80
	C	27
	D	26
Q.45	Which of the following is false?	
	A	Compared to B-tree, B+ -tree has larger fanout
	B	Deletion in B-tree is more complicated than in B+ -tree
	C	B+ -tree has greater depth than corresponding B-tree
	D	Both B-tree and B+ -tree have same search and insertion efficiencies
Q.46	Which one of the following data structures are preferred in database-system implementation?	
	A	AVL tree
	B	B-tree
	C	B+ -tree
	D	Splay tree

Q.47	A B+ -tree of order 3 is generated by inserting 89, 9 and 8. The generated B+ -tree is _____
A	
B	
C	
D	None of These

Prepared by
Prof. Neha Patil
AISSM, Pune