SUBJECT NAME: FUNDAMENTALS OF DATA STRUCTURES

Multiple Choice Questions (MCQ)
LINEAR DATA STRUCTURES – STACKS, QUEUES

1. Process of inserting an element in stack is called _____

a) Create

b) Push

c) Evaluation

d) Pop

Answer: b

2. Process of removing an element from stack is called _____

a) Create

b) Push

c) Evaluation

d) Pop

Answer: d

3. In a stack, if a user tries to remove an element from empty stack it is called _____

a) Underflow

b) Empty collection

c) Overflow

d) Garbage Collection

Answer: a

4. Pushing an element into stack already having five elements and stack size of 5, then stack becomes

a) Overflow

b) Crash

c) Underflow

d) User flow

Answer: a

5. Entries in a stack are “ordered”. What is the meaning of this statement?

a) A collection of stacks is sortable

b) Stack entries may be compared with the ‘<‘ operation

c) The entries are stored in a linked list

d) There is a Sequential entry that is one by one

Answer: d

6. Which of the following is not the application of stack?

a) A parentheses balancing program

b) Tracking of local variables at run time

c) Compiler Syntax Analyzer

d) Data Transfer between two asynchronous process

Answer: d

7. Consider the usual algorithm for determining whether a sequence of parentheses is balanced.

The maximum number of parentheses that appear on the stack AT ANY ONE TIME when the algorithm analyzes: (()(())(())) are:

a) 1

b) 2

c) 3

Answer: c

8. Consider the usual algorithm for determining whether a sequence of parentheses is balanced.

Suppose that you run the algorithm on a sequence that contains 2 left parentheses and 3 right parentheses (in some order).

The maximum number of parentheses that appear on the stack AT ANY ONE TIME during the computation?

a) 1

b) 2

Answer: b

9. What is the value of the postfix expression 6 3 2 4 + – *?

a) 1

b) 40

c) 74

d) -18

Answer: d

10. Here is an infix expression: 4 + 3*(6*3-12). Suppose that we are using the usual stack algorithm to convert the expression from infix to postfix notation.

The maximum number of symbols that will appear on the stack AT ONE TIME during the conversion of this expression?

a) 1

b) 2

c) 3

Answer: d

11. The postfix form of the expression (A+ B)*(C*D- E)*F / G is?

a) AB+ CD*E – FG /**

b) AB + CD* E – F **G /

c) AB + CD* E – *F *G /

d) AB + CDE * – * F *G /

Answer: c

12. The data structure required to check whether an expression contains balanced parenthesis is?

a) Stack

b) Queue

c) Array

d) Tree

Answer: a

13. What data structure would you mostly likely see in a non recursive implementation of a recursive algorithm?

a) Linked List

b) Stack

c) Queue

d) Tree

Answer: b

14. The process of accessing data stored in a serial access memory is similar to manipulating data on a _____

a) Heap

b) Binary Tree

c) Array

d) Stack

Answer: d

15. The postfix form of A*B+C/D is?

a) *AB/CD+

b) AB*CD/+

c) A*BC+/D

d) ABCD+/*

Answer: b

16. Which data structure is needed to convert infix notation to postfix notation?

a) Branch

b) Tree

c) Queue

d) Stack

Answer: d

17. The prefix form of A-B/ (C * D ^ E) is?

a) -/*^ACBDE

b) -ABCD*^DE

c) -A/B*C^DE

d) -A/BC*^DE

Answer: c

18. What is the result of the following operation?

Top (Push (S, X))

a) X

b) X+S

c) S

Answer: a

19. The prefix form of an infix expression (p + q) – (r * t) is?

a) + pq – *rt

b) – +pqr * t

c) – +pq * rt

d) – + * pqrt

Answer: c

20. Which data structure is used for implementing recursion?

a) Queue

b) Stack

c) Array

d) List

Answer: b

21. When an operand is read, which of the following is done?

a) It is placed on to the output

b) It is placed in operator stack

c) It is ignored

d) Operator stack is emptied

Answer: a

22. What should be done when a left parenthesis ‘(‘ is encountered?

a) It is ignored

b) It is placed in the output

c) It is placed in the operator stack

d) The contents of the operator stack is emptied

Answer: c

23. Which of the following is an infix expression?

a) (a+b)*(c+d)

b) ab+c*

c) +ab

d) abc+*

Answer: a

24. What is the time complexity of an infix to postfix conversion algorithm?

a) O(N log N)

b) O(N)

c) O(N2)

d) O(M log N)

Answer: b

25. a) abc*+de*+

b) abc+*de*+

c) a+bc*de+*

d) abc*+(de)*+

Answer: a

26. a) -ab-c

b) ab – c –

c) – -abc

d) -ab-c

Answer: b

27. a) abc^/d-

b) ab/cd^-

c) ab/^cd-

d) abcd^/-

Answer: a

28. Which of the following statement is incorrect with respect to infix to postfix conversion algorithm?

a) operand is always placed in the output

b) operator is placed in the stack when the stack operator has lower precedence

c) parenthesis are included in the output

d) higher and equal priority operators follow the same condition

Answer: c

29. In infix to postfix conversion algorithm, the operators are associated from?

a) right to left

b) left to right

c) centre to left

d) centre to right

Answer: b

30. a) ab*+cd/

b) ab+*cd/

c) abc*+/d

d) abc+*d/

Answer: d

31. a) ab*cdef/^*g-h+

b) abcdef^/*g*h*+

c) abcd*^ed/g*-h*+

d) abc*de^fg/*-*h+

Answer: b

32. a) abc^de-fg+*^*+i-

b) abcde^-fg*+*^h*+i-

c) abcd^e-fgh*+^*+i-

d) ab^-dc*+ef^gh*+i-

Answer: c

33. From the given Expression tree, identify the correct postfix expression from the list of options.

a) ab*cd*+

b) ab*cd-+

c) abcd-*+

Answer: b

34. A linear list of elements in which deletion can be done from one end (front) and insertion can take place only at the other end (rear) is known as a ?

a) Queue

b) Stack

c) Tree

d) Linked list

Answer: a

35. The data structure required for Breadth First Traversal on a graph is?

a) Stack

b) Array

c) Queue

d) Tree

Answer: c

36. A queue follows _____

a) FIFO (First In First Out) principle

b) LIFO (Last In First Out) principle

c) Ordered array

d) Linear tree

Answer: a

37. Circular Queue is also known as _____

a) Ring Buffer

b) Square Buffer

c) Rectangle Buffer

d) Curve Buffer

Answer: a

38. If the elements “A”, “B”, “C” and “D” are placed in a queue and are deleted one at a time, in what order will they be removed?

a) ABCD

b) DCBA

c) DCAB

d) ABDC

Answer: a

39. A data structure in which elements can be inserted or deleted at/from both the ends but not in the middle is?

a) Queue

b) Circular queue

c) Dequeue

d) Priority queue

Answer: c

40. A normal queue, if implemented using an array of size MAX_SIZE, gets full when

a) Rear = MAX_SIZE – 1

b) Front = (rear + 1)mod MAX_SIZE

c) Front = rear + 1

d) Rear = front

Answer: a

41. Queues serve major role in _____

a) Simulation of recursion

b) Simulation of arbitrary linked list

c) Simulation of limited resource allocation

d) Simulation of heap sort

Answer: c

42. Which of the following is not the type of queue?

a) Ordinary queue

b) Single ended queue

c) Circular queue

d) Priority queue

Answer: b

43. With what data structure can a priority queue be implemented?

a) Array

b) List

c) Heap

d) Tree

Answer: d

44. Which of the following is not an application of priority queue?

a) Huffman codes

b) Interrupt handling in operating system

c) Undo operation in text editors

d) Bayesian spam filter

Answer: c

45. What is the time complexity to insert a node based on key in a priority queue?

a) O(nlogn)

b) O(logn)

c) O(n)

d) O(n2)

Answer: c

46. a) Delete the second element in the list

b) Return but not delete the second element in the list

c) Delete the first element in the list

d) Return but not delete the first element in the list

Answer: c

47. What is not a disadvantage of priority scheduling in operating systems?

a) A low priority process might have to wait indefinitely for the CPU

b) If the system crashes, the low priority systems may be lost permanently

c) Interrupt handling

d) Indefinite blocking

Answer: c

48. Which of the following is not an advantage of priority queue?

a) Easy to implement

b) Processes with different priority can be efficiently handled

c) Applications with differing requirements

d) Easy to delete elements in any case

Answer: d

49. What is the time complexity to insert a node based on position in a priority queue?

a) O(nlogn)

b) O(logn)

c) O(n)

d) O(n2)

Answer: c

50. What is a dequeue?

a) A queue with insert/delete defined for both front and rear ends of the queue

b) A queue implemented with a doubly linked list

c) A queue implemented with both singly and doubly linked lists

d) A queue with insert/delete defined for front side of the queue

Answer: a

51. a) Insert at the front end of the dequeue

b) Insert at the rear end of the dequeue

c) Fetch the element at the rear end of the dequeue

d) Fetch the element at the front end of the dequeue

Answer: b

52. What are the applications of dequeue?

a) A-Steal job scheduling algorithm

b) Can be used as both stack and queue

c) To find the maximum of all sub arrays of size k

d) To avoid collision in hash tables

Answer: d

53. a) 10 30 10 15

b) 20 30 40 15

c) 20 30 40 10

d) 10 30 40 15

Answer: d

54. Which of the following properties is associated with a queue?

a) First In Last Out

b) First In First Out

c) Last In First Out

d) Last In Last Out

Answer: b

55. In a circular queue, how do you increment the rear end of the queue?

a) rear++

b) (rear+1) % CAPACITY

c) (rear % CAPACITY)+1

d) rear–

Answer: b

56. What is the term for inserting into a full queue known as?

a) overflow

b) underflow

c) null pointer exception

d) program won’t be compiled

Answer: a

57. What is the time complexity of enqueue operation?

a) O(logn)

b) O(nlogn)

c) O(n)

d) O(1)

Answer: d

58. a) Dequeue

b) Enqueue

c) Return the front element

d) Return the last element

Answer: c

59. What is the need for a circular queue?

a) effective usage of memory

b) easier computations

c) to delete elements based on priority

d) implement LIFO principle in queues

Answer: a

60. What is the space complexity of a linear queue having n elements?

a) O(n)

b) O(nlogn)

c) O(logn)

d) O(1)

Answer: a

61. a) 3 3

b) 3 6

c) 6 6

d) 10 6

Answer: a

Multiple Choice Questions (MCQ)

NON LINEAR DATA STRUCTURES – TREES

1. What is the maximum number of children that a binary tree node can have?

a) 0

b) 1

c) 2

d) 3

Answer: c

2. The following given tree is an example for?

a) Binary tree

b) Binary search tree

c) Fibonacci tree

Answer: a

3. How many common operations are performed in a binary tree?

a) 1

b) 2

c) 3

d) 4

Answer: c

4. What is the traversal strategy used in the binary tree?

a) depth-first traversal

b) breadth-first traversal

c) random traversal

d) Priority traversal

Answer: b

5. How many types of insertion are performed in a binary tree?

a) 1

b) 2

c) 3

d) 4

Answer: b

6. What operation does the following diagram depict?

a) inserting a leaf node

b) inserting an internal node

c) deleting a node with 0 or 1 child

Answer: c

7. How many bits would a succinct binary tree occupy?

a) n+O(n)

b) 2n+O(n)

c) n/2

d) n

Answer: b

8. The average depth of a binary tree is given as?

a) O(N)

b) O(√N)

c) O(N2)

d) O(log N)

Answer: d

9. How many orders of traversal are applicable to a binary tree (In General)?

a) 1

b) 4

c) 2

d) 3

Answer: d

10. If binary trees are represented in arrays, what formula can be used to locate a left child, if the node has an index i?

a) 2i+1

b) 2i+2

c) 2i

d) 4i

Answer: a

11. Using what formula can a parent node be located in an array?

a) (i+1)/2

b) (i-1)/2

c) i/2

d) 2i/2

Answer: b

12. Which of the following properties are obeyed by all three tree – traversals?

a) Left subtrees are visited before right subtrees

b) Right subtrees are visited before left subtrees

c) Root node is visited before left subtree

d) Root node is visited before right subtree

Answer: a

13. Construct a binary tree using the following data.

The preorder traversal of a binary tree is 1, 2, 5, 3, 4. The inorder traversal of the same binary tree is 2, 5, 1, 4, 3.

a)

b)

c)

Answer: d

14. For the tree below, write the pre-order traversal.

a) 2, 7, 2, 6, 5, 11, 5, 9, 4

b) 2, 7, 5, 2, 6, 9, 5, 11, 4

c) 2, 5, 11, 6, 7, 4, 9, 5, 2

Answer: a

15. For the tree below, write the post-order traversal.

a) 2, 7, 2, 6, 5, 11, 5, 9, 4

b) 2, 7, 5, 2, 6, 9, 5, 11, 4

c) 2, 5, 11, 6, 7, 4, 9, 5, 2

Answer: c

16. What is the time complexity of pre-order traversal in the iterative fashion?

a) O(1)

b) O(n)

c) O(logn)

d) O(nlogn)

Answer: b

17. What is the space complexity of the post-order traversal in the recursive fashion? (d is the tree depth and n is the number of nodes)

a) O(1)

b) O(nlogd)

c) O(logd)

d) O(d)

Answer: d

18. To obtain a prefix expression, which of the tree traversals is used?

a) Level-order traversal

b) Pre-order traversal

c) Post-order traversal

d) In-order traversal

Answer: b

19. Consider the following data. The pre order traversal of a binary tree is A, B, E, C, D. The in order traversal of the same binary tree is B, E, A, D, C. The level order sequence for the binary tree is _____

a) A, C, D, B, E

b) A, B, C, D, E

c) A, B, C, E, D

d) D, B, E, A, C

Answer: b

20. Consider the following data and specify which one is Preorder Traversal Sequence, Inorder and Postorder sequences.

S1: N, M, P, O, Q

S2: N, P, Q, O, M

S3: M, N, O, P, Q

a) S1 is preorder, S2 is inorder and S3 is postorder

Answer: c

21. What is the possible number of binary trees that can be created with 3 nodes, giving the sequence N, M, L when traversed in post-order.

a) 15

b) 3

c) 5

d) 8

Answer: c

22. The post-order traversal of a binary tree is O P Q R S T. Then possible pre-order traversal will be _____

a) T Q R S O P

b) T O Q R P S

c) T Q O P S R

d) T Q O S P R

Answer: c

23. A binary search tree contains values 7, 8, 13, 26, 35, 40, 70, 75. Which one of the following is a valid post-order sequence of the tree provided the pre-order sequence as 35, 13, 7, 8, 26, 70, 40 and 75?

a) 7, 8, 26, 13, 75, 40, 70, 35

b) 26, 13, 7, 8, 70, 75, 40, 35

c) 7, 8, 13, 26, 35, 40, 70, 75

d) 8, 7, 26, 13, 40, 75, 70, 35

Answer: d

24. Which of the following pair's traversals on a binary tree can build the tree uniquely?

a) post-order and pre-order

b) post-order and in-order

c) post-order and level order

d) level order and preorder

Answer: b

25. A full binary tree can be generated using _____

a) post-order and pre-order traversal

b) pre-order traversal

c) post-order traversal

d) in-order traversal

Answer: a

26. The maximum number of nodes in a tree for which post-order and pre-order traversals may be equal is _____

a) 3

b) 1

c) 2

d) any number

Answer: b

27. The pre-order and in-order are traversals of a binary tree are T M L N P O Q and L M N T O P Q. Which of following is post-order traversal of the tree?

a) L N M O Q P T

b) N M O P O L T

c) L M N O P Q T

d) O P L M N Q T

Answer: a

28. Find the postorder traversal of the binary tree shown below.

a) P Q R S T U V W X

b) W R S Q P V T U X

c) S W T Q X U V R P

Answer: c

29. For the tree below, write the in-order traversal.

a) 6, 2, 5, 7, 11, 2, 5, 9, 4

b) 6, 5, 2, 11, 7, 4, 9, 5, 2

c) 2, 7, 2, 6, 5, 11, 5, 9, 4

Answer: a

30. For the tree below, write the level-order traversal.

a) 2, 7, 2, 6, 5, 11, 5, 9, 4

b) 2, 7, 5, 2, 11, 9, 6, 5, 4

c) 2, 5, 11, 6, 7, 4, 9, 5, 2

Answer: b

31. What is the space complexity of the in-order traversal in the recursive fashion? (d is the tree depth and n is the number of nodes)

a) O(1)

b) O(nlogd)

c) O(logd)

d) O(d)

Answer: d

32. What is the time complexity of level order traversal?

a) O(1)

b) O(n)

c) O(logn)

d) O(nlogn)

Answer: b

33. Which of the following graph traversals closely imitates level order traversal of a binary tree?

a) Depth First Search

b) Breadth First Search

c) Depth & Breadth First Search

d) Binary Search

Answer: b

34. In a binary search tree, which of the following traversals would print the numbers in the ascending order?

a) Level-order traversal

b) Pre-order traversal

c) Post-order traversal

d) In-order traversal

Answer: d

35. The number of edges from the root to the node is called_____of the tree.

a) Height

b) Depth

c) Length

d) Width

Answer: b

36. The number of edges from the node to the deepest leaf is called_____of the tree.

a) Height

b) Depth

c) Length

d) Width

Answer: a

37. What is a full binary tree?

a) Each node has exactly zero or two children

b) Each node has exactly two children

c) All the leaves are at the same level

d) Each node has exactly one or two children

Answer: a

38. What is a complete binary tree?

a) Each node has exactly zero or two children

b) A binary tree, which is completely filled, with the possible exception of the bottom level, which is filled from right to left

c) A binary tree, which is completely filled, with the possible exception of the bottom level, which is filled from left to right

d) A tree In which all nodes have degree 2

Answer: c

39. What is the average case time complexity for finding the height of the binary tree?

a) h = O(loglogn)

b) h = O(nlogn)

c) h = O(n)

d) h = O(log n)

Answer: d

40. Which of the following is not an advantage of trees?

a) Hierarchical structure

b) Faster search

c) Router algorithms

d) Undo/Redo operations in a notepad

Answer: d

41. In a full binary tree if number of internal nodes is I, then number of leaves L are?

a) L = 2*I

b) L = I + 1

c) L = I – 1

d) L = 2*I – 1

Answer: b

42. In a full binary tree if number of internal nodes is I, then number of nodes N are?

a) N = 2*I

b) N = I + 1

c) N = I – 1

d) N = 2*I + 1

Answer: d

43. In a full binary tree if there are L leaves, then total number of nodes N are?

a) N = 2*L

b) N = L + 1

c) N = L – 1

d) N = 2*L – 1

Answer: d

44. Which of the following is incorrect with respect to binary trees?

a) Let T be a binary tree. For every $k \geq 0$, there are no more than 2k nodes in level k

b) Let T be a binary tree with λ levels. Then T has no more than $2\lambda - 1$ nodes

c) Let T be a binary tree with N nodes. Then the number of levels is at least ceil(log (N + 1))

d) Let T be a binary tree with N nodes. Then the number of levels is at least floor(log (N + 1))

Answer: d

45. Construct a binary tree by using postorder and inorder sequences given below.

Inorder: N, M, P, O, Q

Postorder: N, P, Q, O, M

a)

b)

Answer: d

46. Construct a binary search tree by using postorder sequence given below.

Postorder: 2, 4, 3, 7, 9, 8, 5.

a)

b)

c)

Answer: b

47. Construct a binary tree using inorder and level order traversal given below.

Inorder Traversal: 3, 4, 2, 1, 5, 8, 9

Level Order Traversal: 1, 4, 5, 9, 8, 2, 3

a)

b)

Answer: a

48. Which of the following is false about a binary search tree?

a) The left child is always lesser than its parent

b) The right child is always greater than its parent

c) The left and right sub-trees should also be binary search trees

d) In order sequence gives decreasing order of elements

Answer: d

49. What is the speciality about the inorder traversal of a binary search tree?

a) It traverses in a non increasing order

b) It traverses in an increasing order

c) It traverses in a random fashion

d) It traverses based on priority of the node

Answer: b

50. a) Preorder traversal

b) Inorder traversal

c) Postorder traversal

d) Level order traversal

Answer: c

51. a) Preorder traversal

b) Inorder traversal

c) Postorder traversal

d) Level order traversal

Answer: a

52. What are the worst case and average case complexities of a binary search tree?

a) O(n), O(n)

b) O(logn), O(logn)

c) O(logn), O(n)

d) O(n), O(logn)

Answer: d

53. What are the conditions for an optimal binary search tree and what is its advantage?

a) The tree should not be modified and you should know how often the keys are accessed, it improves the lookup cost

b) You should know the frequency of access of the keys, improves the lookup time

c) The tree can be modified and you should know the number of elements in the tree before hand, it improves the deletion time

d) The tree should be just modified and improves the lookup time

Answer: a

54. Construct a binary search tree with the below information.

The preorder traversal of a binary search tree 10, 4, 3, 5, 11, 12.

a)

b)

c)

Answer: c

55. Which of the following is not the self balancing binary search tree?

a) AVL Tree

b) 2-3-4 Tree

c) Red – Black Tree

d) Splay Tree

Answer: b

56. The binary tree sort implemented using a self – balancing binary search tree takes_____time is worst case.

a) O(n log n)

b) O(n)

c) O(n2)

d) O(log n)

Answer: a

57. An AVL tree is a self – balancing binary search tree, in which the heights of the two child sub trees of any node differ by _____

a) At least one

b) At most one

c) Two

d) At most two

Answer: b

58. Associative arrays can be implemented using _____

a) B-tree

b) A doubly linked list

c) A single linked list

d) A self balancing binary search tree

Answer: d

59. Which of the following is a self – balancing binary search tree?

a) 2-3 tree

b) Threaded binary tree

c) AA tree

d) Treap

Answer: c

60. A self – balancing binary search tree can be used to implement _____

a) Priority queue

b) Hash table

c) Heap sort

d) Priority queue and Heap sort

Answer: a

61. In which of the following self – balancing binary search tree the recently accessed element can be accessed quickly?

a) AVL tree

b) AA tree

c) Splay tree

d) Red – Black tree

Answer: c

62. The minimum height of self balancing binary search tree with n nodes is _____

a) log2(n)

b) n

c) 2n + 1

d) 2n – 1

Answer: a

63. What is an AVL tree?

a) a tree which is balanced and is a height balanced tree

b) a tree which is unbalanced and is a height balanced tree

c) a tree with three children

d) a tree with atmost 3 children

Answer: a

64. Why we need to a binary tree which is height balanced?

a) to avoid formation of skew trees

b) to save memory

c) to attain faster memory access

d) to simplify storing

Answer: a

65. Which of the below diagram is following AVL tree property?

i.

ii.

a) only i

b) only i and ii

Answer: b

66. What is the maximum height of an AVL tree with p nodes?

a) p

b) log(p)

c) log(p)/2

d) p⁄2

Answer: b

67. Given an empty AVL tree, how would you construct AVL tree when a set of numbers are given without performing any rotations?

a) just build the tree with the given input

b) find the median of the set of elements given, make it as root and construct the tree

c) use trial and error

d) use dynamic programming to build the tree

Answer: b

68. What maximum difference in heights between the leafs of a AVL tree is possible?

a) log(n) where n is the number of nodes

b) n where n is the number of nodes

c) 0 or 1

d) atmost 1

Answer: a

69. What is missing?

a) Height(w-left), x-height

b) Height(w-right), x-height

c) Height(w-left), x

d) Height(w-left)

Answer: a

70. Why to prefer red-black trees over AVL trees?

a) Because red-black is more rigidly balanced

b) AVL tree store balance factor in every node which costs space

c) AVL tree fails at scale

d) Red black is more efficient

Answer: b

71. Which of the following is the most widely used external memory data structure?

a) AVL tree

b) B-tree

c) Red-black tree

d) Both AVL tree and Red-black tree

Answer: b

72. B-tree of order n is a order-n multiway tree in which each non-root node contains _____

a) at most (n – 1)/2 keys

b) exact (n – 1)/2 keys

c) at least 2n keys

d) at least (n – 1)/2 keys

Answer: d

73. A B-tree of order 4 and of height 3 will have a maximum of_____keys.

a) 255

b) 63

c) 127

d) 188

Answer: a

74. Five node splitting operations occurred when an entry is inserted into a B-tree. Then how many nodes are written?

a) 14

b) 7

c) 11

d) 5

Answer: c

75. trees are B-trees of order 4. They are an isometric of_____trees.

a) AVL

b) AA

c) 2-3

d) Red-Black

Answer: d

76. Figure shown below is B-tree of order 5. What is the result of deleting 130 from the tree?

a)

b)

c)

Answer: c

77. What is the best case height of a B-tree of order n and which has k keys?

a) logn (k+1) – 1

b) nk

c) logk (n+1) – 1

d) klogn

Answer: a

78. Which of the following is true?

a) larger the order of B-tree, less frequently the split occurs

b) larger the order of B-tree, more frequently the split occurs

c) smaller the order of B-tree, more frequently the split occurs

d) smaller the order of B-tree, less frequently the split occurs

Answer: a

79. In a max-heap, element with the greatest key is always in the which node?

a) Leaf node

b) First node of left sub tree

c) root node

d) First node of right sub tree

Answer: c

80. What is the complexity of adding an element to the heap.

a) O(log n)

b) O(h)

c) O(log n) & O(h)

d) O(n)

Answer: c

81. The worst case complexity of deleting any arbitrary node value element from heap is _____

a) O(logn)

b) O(n)

c) O(nlogn)

d) O(n2)

Answer: a

82. Heap can be used as _____

a) Priority queue

b) Stack

c) A decreasing order array

d) Normal Array

Answer: a

83.

If we implement heap as min-heap, deleting root node (value 1)from the heap. What would be the value of root node after second iteration if leaf node (value 100) is chosen to replace the root at start.

a) 2

b) 100

c) 17

Answer: a

84.

If we implement heap as maximum heap , adding a new node of value 15 to the left most node of right subtree . What value will be at leaf nodes of the right subtree of the heap.

a) 15 and 1

b) 25 and 1

c) 3 and 1

Answer: a

85. An array consists of n elements. We want to create a heap using the elements. The time complexity of building a heap will be in order of

a) O(n*n*logn)

b) O(n*logn)

c) O(n*n)

d) O(n *logn *logn)

Answer: b