UNIT 1 : Basic Concept and Introduction to Data structure

1. The word ____________comes from the name of a Persian mathematician Abu Ja'far Mohammed ibn-i Musa al Khowarizmi.
a) Flowchart
b) Flow
c) Algorithm
d) Syntax
Answer: c

2. In computer science, algorithm refers to a special method usable by a computer for the solution to a problem.
a) True
b) False
c. Maybe
d. None of the above
Answer: a

3. Which of the following is incorrect? Algorithms can be represented:
a) as pseudo codes
b) as syntax
c) as programs
d) as flowcharts
Answer: b.

4. When an algorithm is written in the form of a programming language, it becomes a _________
a) Flowchart
b) Program
c) Pseudo code
d) Syntax
Answer: b

5. Any algorithm is a program.
a) True
b) False
c. Maybe
d. None of the above
Answer: b

6. In computer science, algorithm refers to a pictorial representation of a flowchart.
a) True
b) False
c. Maybe
d. None of the above

Answer: b

7. The process of drawing a flowchart for an algorithm is called __________
a) Performance
b) Evaluation
c) Algorithmic Representation
d) Flowcharting
Answer: d

8. Which of the following header files must necessarily be included to use dynamic memory allocation functions?
a) stdlib.h
b) stdio.h
c) memory.h
d) dos.h
Answer: a

9. In the function malloc(), each byte of allocated space is initialized to zero.
a) True
b) False
c. Maybe
d. None of the above
Answer: b

10. Which of the following functions allocates multiple blocks of memory, each block of the same size?
a) malloc()
b) realloc()
c) calloc()
d) free()
Answer: c

11. A condition where in memory is reserved dynamically but not accessible to any of the programs is called _____________
a) Memory leak
b) Dangling pointer
c) Frozen memory
d) Pointer leak
Answer: a

12. The incorrect statement with respect to dangling pointers is ___________
a) Pointer pointing to non-existent memory location is called dangling pointer
b) When a dynamically allocated pointer references the original memory after it has been freed, a dangling pointer arises
c) If memory leak occurs, it is mandatory that a dangling pointer arises

d) Dangling pointer may result in segmentation faults and potential security risks
Answer: c

13. In the function realloc(), if the new size of the memory block is larger than the old size, then the added memory ___________
a) is initialized to junk values
b) is initialized to zero
c) results in an error
d) is not initialized
Answer: d

14. The free() function frees the memory state pointed to by a pointer and returns ___________
a) the same pointer
b) the memory address
c) no value
d) an integer value
Answer: c

15. The number of arguments taken as input which allocating memory dynamically using malloc() is ___________
a) 0
b) 1
c) 2
d) 3
Answer: b

16. Suppose we have a one dimensional array, named 'x', which contains 10 integers. Which of the following is the correct way to allocate memory dynamically to the array 'x' using malloc()?
a) x=(int*)malloc(10);
b) x=(int*)malloc(10,sizeof(int));
c) x=malloc(int 10,sizeof(int));
d) x=(int*)malloc(10*sizeof(int));
Answer: d

17. If malloc() and calloc() are not type casted, the default return type is ___________
a) void*
b) void**
c) int*
d) char*
Answer: a

18. When the pointer is NULL, then the function realloc is equivalent to the function ___________
a) malloc
b) calloc
c) free

d) alloc
Answer: a

19. Garbage collector frees the programmer from worrying about ___________
a) Dangling pointers
b) Creating new objects
c) Memory leak
d) Segmentation errors
Answer: c

20. If the space in memory allocated by malloc is not sufficient, then an allocation fails and returns ___________
a) NULL pointer
b) Zero
c) Garbage value
d) The number of bytes available
Answer: a

21. Among 4 header files, which should be included to use the memory allocation functions like malloc(), calloc(), realloc() and free()?
a) #include<string.h>
b) #include<stdlib.h>
c) #include<memory.h>
d) Both b and c
Answer: b

22. Which of the following is/are true
a) calloc() allocates the memory and also initializes the allocates memory to zero, while memory allocated using malloc() has random data)
b) malloc() and memset() can be used to get the same effect as calloc()
c) Both malloc() and calloc() return 'void *' pointer
d) all of the above
Answer: d

23. Which of the following statement is correct prototype of the malloc() function in c ?
a) int* malloc(int);
b) Char* malloc(char);
c) unsigned int* malloc(unsigned int);
d) void* malloc(size_t);
Answer: d

24. Specify the 2 library functions to dynamically allocate memory?
a) malloc() and memalloc()

b) alloc() and memalloc()
c) malloc() and calloc()
d) memalloc() and faralloc()
Answer: c

25. malloc() returns a float pointer if memory is allocated for storing float's and a double pointer if memory is allocated for storing double's. a)
a) TRUE
b) FaLSE
c) May Be
d) Can't Say
Answer: b

26. malloc() allocates memory from the heap and not from the stack.
a) TRUE
b) FaLSE
c) May Be
d) Can't Say
Answer: a

27. What is wild pointer?
a) Pointer which is wild in nature
b) Pointer which has no value.
c) Pointer which is not initialized
d) None
Answer: c

28. address stored in the pointer variable is of type __________.
a) Integer
b) Float
c) array
d) Character
Answer: a

29. In order to fetch the address of the variable we write preceding _________ sign before variable name.
a) Percent(%)
b) Comma(,)
c) ampersand(&)
d) asteric(*)
Answer: c

30.Space complexity of an algorithm is the maximum amount of _______ required by it during execution.
a)Time

b.Operations
c.Memory space
d.None of the above
Answer c

31.Frequently, the memory space required by an algorithm is a multiple of the size of input. State if the statement is True or False or Maybe.
a) True
b. False
c. Maybe
d. None of the above
Answer a

32an algorithm performs lesser number of operations when the size of input is small, but performs more operations when the size of input gets larger. State if the statement is True or False or Maybe.
a) True
b. False
c. Maybe
d. None of the above
Answer a

33.In the analysis of algorithms, what plays an important role?
a) Text analysis
b. Growth factor
c. Time
d. None of the above
Answer b

34 To verify whether a function grows faster or slower than the other function, we have some asymptotic or mathematical notations, which is__________.
a) Big Omega Ω (f)
b. Big Theta θ (f)
c. Big Oh O (f)
d. all of the above
Answer d
35. an algorithm that indicates the amount of temporary storage required for running the algorithm, i.e., the amount of memory needed by the algorithm to run to completion is termed as_____.
a) Big Theta θ (f)

b. Space complexity
c. Big Oh O (f)
d. None of the above
Answer b

36. The amount of time the computer needs to run to completion is known as_____.
a) Space complexity
b. Time complexity
c. Recursive function
d. None of the above
Answer b

37. ___________algorithm is one which utilizes minimum processor time and requires minimum memory space during its execution.
a) Best
b. Efficient
c. Both (a) and (b)
d. None of the above
Answer c

38. What is (void*)0?
a)Representation of NULL pointer
b)Representation of void pointer
c)Error
d)None of above
Answer: a

39. If a variable is a pointer to a structure, then which of the following operator is used to access data members of the structure through the pointer variable?
a) .
b) &
c) *
d) ->
Answer: d

40. a pointer is
a) a keyword used to create variables
b) a variable that stores address of an instruction

c) a variable that stores address of other variable

d) all of the above

Answer: c

41.The operator used to get value at address stored in a pointer variable is

a) *

b) &

c) &&

d) ||

Answer: a

42. Choose the correct option about abstract data type(aDT).

a) an abstract data type is a model of a certain kind of data structure.

b. In abstract data type we know what a specific data type can do, but how it actually does it is hidden.

c. aDT is user defined type.

d. all of the above.

Answer: d

43. Which of these best describes an array?

a) a data structure that shows a hierarchical behaviour

b) Container of objects of similar types

c) arrays are immutable once initialised

d) array is not a data structure

View answer

Answer: b

44. How do you initialize an array in C?

a) intarr[3] = (1,2,3);

b) intarr(3) = {1,2,3};

c) intarr[3] = {1,2,3};

d) intarr(3) = (1,2,3);

View answer

Answer: c

45. How do you instantiate an array in Java?

a) intarr[] = new int(3);

b) intarr[];

c) intarr[] = new int[3];

d) intarr() = new int(3);

Answer: c

46. Which of the following is a correct way to declare a multidimensional array in Java?
a) int[] arr;
b) intarr[[]];
c) int[][]arr;
d) int[[]] arr;
Answer: c

47. a mathematical-model with a collection of operations defined on that model is called
a) Data Structure
b) abstract Data Type
c) Primitive Data Type
d) algorithm
Answer: b

48. Representation of data structure in memory is known as:
a) Recursive
b) abstract data type
c) Storage structure
d) File structure
Answer: b

49. Which of the following abstract data types can be used to represent a many to many relation?
a) Tree
b) Plex
c) Graph
d) Both (b) and (c)
Answer: d

50. When does the arrayIndexOutOfBounds Exception occur?
a) Compile-time
b) Run-time
c) Not an error
d) Not an exception at all
Answer: b

UNIT 2 : Linear Data Structure

1. What is an external sorting algorithm?
a) Algorithm that uses tape or disk during the sort
b) Algorithm that uses main memory during the sort
c) Algorithm that involves swapping
d) Algorithm that are considered 'in place'
Answer: a

2. What is an internal sorting algorithm?
a) Algorithm that uses tape or disk during the sort
b) Algorithm that uses main memory during the sort
c) Algorithm that involves swapping
d) Algorithm that are considered 'in place'
Answer: b

3. What is the worst case complexity of bubble sort?
a) O(nlogn)
b) O(logn)
c) O(n)
d) O(n^2)
Answer: d

4. What is the average case complexity of bubble sort?
a) O(nlogn)
b) O(logn)
c) O(n)
d) O(n^2)
Answer: d

5. Which of the following is not an advantage of optimised bubble sort over other sorting techniques in case of sorted elements?
a) It is faster
b) Consumes less memory
c) Detects whether the input is already sorted
d) Consumes less time
Answer: c

6. The given array is arr = {1, 2, 4, 3}. Bubble sort is used to sort the array elements. How many iterations will be done to sort the array?
a) 4
b) 2
c) 1

d) 0
Answer: a

7. What is the best case efficiency of bubble sort in the improvised version?
a) O(nlogn)
b) O(logn)
c) O(n)
d) O(n^2)
Answer: c

8. The given array is arr = {1,2,4,3}. Bubble sort is used to sort the array elements. How many iterations will be done to sort the array with improvised version?
a) 4
b) 2
c) 1
d) 0
Answer: b

9. Merge sort uses which of the following technique to implement sorting?
a) backtracking
b) greedy algorithm
c) divide and conquer
d) dynamic programming
Answer: c

10. What is the average case time complexity of merge sort?
a) O(n log n)
b) O(n^2)
c) O(n^2 log n)
d) O(n log n^2)
Answer: a

11. What is the auxiliary space complexity of merge sort?
a) O(1)
b) O(log n)
c) O(n)
d) O(n log n)
Answer: c

12. Which of the following step is taken after finding an element having value greater than the element being searched?
a) linear search takes place in the forward direction
b) linear search takes place in the backward direction
c) binary search takes place in the forward direction

d) binary search takes place in a backward direction
Answer: b

13. What is the worst case time complexity of merge sort?
a) O(n log n)
b) O(n^2)
c) O(n^2 log n)
d) O(n log n^2)
Answer: a

14. Which of the following method is used for sorting in merge sort?
a) merging
b) partitioning
c) selection
d) exchanging
Answer: a

15. What will be the best case time complexity of merge sort?
a) O(n log n)
b) O(n^2)
c) O(n^2 log n)
d) O(n log n^2)
Answer: a

16. Which of the following is not a variant of merge sort?
a) in-place merge sort
b) bottom up merge sort
c) top down merge sort
d) linear merge sort
Answer: d

17. Choose the incorrect statement about merge sort from the following?
a) it is a comparison based sort
b) it is an adaptive algorithm
c) it is not an in place algorithm
d) it is stable algorithm
Answer: b

18. Which of the following is not in place sorting algorithm?
a) merge sort
b) quick sort
c) heap sort
d) insertion sort

Answer: a

19. Which of the following is not a stable sorting algorithm?
a) Quick sort
b) Cocktail sort
c) Bubble sort
d) Merge sort
Answer: a

20. Which of the following stable sorting algorithm takes the least time when applied to an almost sorted array?
a) Quick sort
b) Insertion sort
c) Selection sort
d) Merge sort
Answer: d

21. Which of the following searching algorithm is fastest?
a) jump search
b) binary search
c) linear search
d) all are equally fast
Answer: b

22. Which of the following sorting algorithm makes use of merge sort?
a) tim sort
b) intro sort
c) bogo sort
d) quick sort
Answer: a

23. Which of the following sorting algorithm does not use recursion?
a) quick sort
b) merge sort
c) heap sort
d) bottom up merge sort
Answer: d

24. What is the advantage of bubble sort over other sorting techniques?
a) It is faster
b) Consumes less memory
c) Detects whether the input is already sorted
d) All of the mentioned
Answer: c

25. The given array is arr = {1,2,4,3}. Bubble sort is used to sort the array elements. How many iterations will be done to sort the array?

a) 4
b) 2
c) 1
d) 0

Answer: a

26. What is the best case complexity of QuickSort?

a) O(nlogn)
b) O(logn)
c) O(n)
d) O(n2)

Answer: a

27. What is the average case complexity of selection sort?

a) O(nlogn)
b) O(logn)
c) O(n)
d) O(n2)

Answer: d

28. What is the disadvantage of selection sort?

a) It requires auxiliary memory
b) It is not scalable
c) It can be used for small keys
d) None of the mentioned

Answer: b

29. What is the worst case complexity of bubble sort?

a) O(nlogn)
b) O(logn)
c) O(n)
d) O(n2)

Answer: d

30. What is the worst case complexity of selection sort?

a) O(nlogn)

b) O(logn)

c) O(n)

d) O(n2)

Answer: d

31. Where is linear searching used?
a) When the list has only a few elements
b) When performing a single search in an unordered list
c) Used all the time
d) When the list has only a few elements and When performing a single search in an unordered list
Answer: d

32. What is the best case for linear search?
a) O(nlogn)
b) O(logn)
c) O(n)
d) O(1)
Answer: d

33. What is the worst case for linear search?
a) O(nlogn)
b) O(logn)
c) O(n)
d) O(1)
Answer: c

34. What is the best case and worst case complexity of ordered linear search?
a) O(nlogn), O(logn)
b) O(logn), O(nlogn)
c) O(n), O(1)
d) O(1), O(n)
Answer: d

35. Which of the following is a disadvantage of linear search?
a) Requires more space
b) Greater time complexities compared to other searching algorithms
c) Not easy to understand
d) Not easy to implement
Answer: b

36. Is there any difference in the speed of execution between linear serach(recursive) vs linear search(lterative)?
a) Both execute at same speed
b) Linear search(recursive) is faster

c) Linear search(Iterative) is faster
d) Cant be said
Answer: c

37. Is the space consumed by the linear search(recursive) and linear search(iterative) same?
a) No, recursive algorithm consumes more space
b) No, recursive algorithm consumes less space
c) Yes
d) Nothing can be said
Answer: a

38. What is the worst case runtime of linear search(recursive) algorithm?
a) O(n)
b) O(logn)
c) O(n^2)
d) O(nx)
Answer: a

39. Linear search(recursive) algorithm used in _____________
a) When the size of the dataset is low
b) When the size of the dataset is large
c) When the dataset is unordered
d) Never used
Answer: a

40. The array is as follows: 1,2,3,6,8,10. At what time the element 6 is found? (By using linear search(recursive) algorithm)
a) 4th call
b) 3rd call
c) 6th call
d) 5th call
Answer: a

41. The array is as follows: 1,2,3,6,8,10. Given that the number 17 is to be searched. At which call it tells that there’s no such element? (By using linear search(recursive) algorithm)
a) 7th call
b) 9th call
c) 17th call
d) The function calls itself infinite number of times
Answer: a

42. What is the best case runtime of linear search(recursive) algorithm on an ordered set of elements?
a) O(1)
b) O(n)

c) O(logn)
d) O(nx)
Answer: a

43. Can linear search recursive algorithm and binary search recursive algorithm be performed on an unordered list?
a) Binary search can't be used
b) Linear search can't be used
c) Both cannot be used
d) Both can be used
Answer: a

44. What is the recurrence relation for the linear search recursive algorithm?
a) T(n-2)+c
b) 2T(n-1)+c
c) T(n-1)+c
d) T(n+1)+c
Answer: c

45. What is the advantage of recursive approach than an iterative approach?
a) Consumes less memory
b) Less code and easy to implement
c) Consumes more memory
d) More code has to be written
Answer: b

46. Given an input arr = {2,5,7,99,899}; key = 899; What is the level of recursion?
a) 5
b) 2
c) 3
d) 4
Answer: c

47. Given an array arr = {45,77,89,90,94,99,100} and key = 99; what are the mid values(corresponding array elements) in the first and second levels of recursion?
a) 90 and 99
b) 90 and 94
c) 89 and 99
d) 89 and 94
Answer: a

48. What is the worst case complexity of binary search using recursion?
a) O(nlogn)
b) O(logn)

c) O(n)
d) O(n^2)
Answer: b

49. What is the average case time complexity of binary search using recursion?
a) O(nlogn)
b) O(logn)
c) O(n)
d) O(n^2)
Answer: b

50. Which of the following is not an application of binary search?
a) To find the lower/upper bound in an ordered sequence
b) Union of intervals
c) Debugging
d) To search in unordered list
Answer: d

51. Binary Search can be categorized into which of the following?
a) Brute Force technique
b) Divide and conquer
c) Greedy algorithm
d) Dynamic programming
Answer: b

52. Given an array arr = {5,6,77,88,99} and key = 88; How many iterations are done until the element is found?
a) 1
b) 3
c) 4
d) 2
Answer: d

53. Given an array arr = {45,77,89,90,94,99,100} and key = 100; What are the mid values(corresponding array elements) generated in the first and second iterations?
a) 90 and 99
b) 90 and 100
c) 89 and 94
d) 94 and 99
Answer: a

54. What is the time complexity of binary search with iteration?
a) O(nlogn)
b) O(logn)
c) O(n)

d) O(n^2)
Answer: b

55. In which of the cases uniform binary search fails compared to binary search?
a) A table lookup is generally faster than an addition and a shift
b) Many searches will be performed on the same array
c) Many searches will be performed on several arrays of the same length
d) Complexity of code
Answer: d

56. Given delta[4] is a global array and number of elements in the sorted array is 10, what are the values in the delta array?
a) 4, 3, 1, 0
b) 5, 3, 1, 0
c) 4, 2, 1, 1
d) 5, 2, 1, 1
Answer: b

57. What is the time complexity of uniform binary search?
a) O(nlogn)
b) O(logn)
c) O(n)
d) O(n^2)
Answer: b

58. Given, arr = {1,3,5,6,7,9,14,15,17,19} key = 17 and delta = {5,3,1,0}
How many key comparisons are made?(exclude the comparison used to decide the left or right sub array)
a) 4
b) 3
c) 5
d) 6
Answer: b

Unit 3 : Linked List

1. A linear collection of data elements where the linear node is given by means of pointer is called?
   a) Linked list
   b) Node list
   c) Primitive list
   d) Unordered list

2. Consider an implementation of unsorted singly linked list. Suppose it has its representation with a head pointer only.
   Given the representation, which of the following operation can be implemented in O(1) time?
   i) Insertion at the front of the linked list
   ii) Insertion at the end of the linked list
   iii) Deletion of the front node of the linked list
   iv) Deletion of the last node of the linked list
   a) I and II
   b) I and III
   c) I, II and III
   d) I, II and IV

3. In linked list each node contain minimum of two fields. One field is data field to store the data second field is?
   a) Pointer to character
   b) Pointer to integer
   c) Pointer to node
   d) Node

4. What would be the asymptotic time complexity to add a node at the end of singly linked list, if the pointer is initially pointing to the head of the list?
   a) O(1)
   b) O(n)
   c) θ(n)
   d) θ(1)

5. What would be the asymptotic time complexity to insert an element at the front of the linked list (head is known)?
   a) O(1)
   b) O(n)
   c) O(n^2)
   d) O(n^3)

6. What would be the asymptotic time complexity to find an element in the linked list?
   a) O(1)

b) O(n)
c) $O(n^2)$
d) $O(n^4)$

7. What would be the asymptotic time complexity to insert an element at the second position in the linked list?
a) O(1)
b) O(n)
c) $O(n^2)$
d) $O(n^3)$

8. The concatenation of two list can performed in O(1) time. Which of the following variation of linked list can be used?
a) Singly linked list
b) Doubly linked list
c) Circular doubly linked list
d) Array implementation of list

9. What kind of linked list is best to answer question like “What is the item at position n?”
a) Singly linked list
b) Doubly linked list
c) Circular linked list
d) Array implementation of linked list

10. Linked lists are not suitable to for the implementation of?
a) Insertion sort
b) Radix sort
c) Polynomial manipulation
d) Binary search

11. Linked list is considered as an example of ___________ type of memory allocation.
a) **Dynamic**
b) Static
c) Compile time
d) Heap

12. In Linked List implementation, a node carries information regarding ___________
a) Data
b) **Link**
c) Data and Link
d) Node

13. Linked list data structure offers considerable saving in _____________
a) Computational Time
b) Space Utilization
c) **Space** Utilization and Computational Time
d) Speed Utilization

14. Which of the following points is/are not true about Linked List data structure when it is compared with array?
a) Arrays have better cache locality that can make them better in terms of performance
b) It is easy to insert and delete elements in Linked List
c) Random access is not allowed in a typical implementation of Linked Lists
d) **Access** of elements in linked list takes less time than compared to arrays

15. Which of the following sorting algorithms can be used to sort a random linked list with minimum time complexity?
a) Insertion Sort
b) Quick Sort
c) Heap Sort
d) **Merge** Sort

16. In the worst case, the number of comparisons needed to search a singly linked list of length n for a given element is
a) log 2 n
b) $^{n}/_{2}$
c) log 2 n – 1
d) n

17. Given pointer to a node X in a singly linked list. Only one pointer is given, pointer to head node is not given, can we delete the node X from given linked list?
a) **Possible** if X is not last node
b) Possible if size of linked list is even
c) Possible if size of linked list is odd
d) Possible if X is not first node

18. You are given pointers to first and last nodes of a singly linked list, which of the following operations are dependent on the length of the linked list?
a) Delete the first element
b) Insert a new element as a first element
c) **Delete** the last element of the list
d) Add a new element at the end of the list

19. In the worst case, the number of comparisons needed to search a singly linked list of length n for a given element is

a) log2 n
b) $^{n}/_{2}$
c) log2 n – 1
d) n

20. Which of the following is not a disadvantage to the usage of array?
a) Fixed size
b) There are chances of wastage of memory space if elements inserted in an array are lesser than the allocated size
c) Insertion based on position
d) **Accessing** elements at specified positions

21. What is the time complexity of inserting at the end in dynamic arrays?
a) O(1)
b) O(n)
c) O(logn)
d) **Either** O(1) or O(n)

22. What is the time complexity to count the number of elements in the linked list?
a) O(1)
b) O(n)
c) O(logn)
d) O(n^2)

23. What is the space complexity for deleting a linked list?
a) O(1)
b) O(n)
c) Either O(1) or O(n)
d) O(logn)

24. Which of the following is false about a doubly linked list?
a) We can navigate in both the directions
b) It requires more space than a singly linked list
c) The insertion and deletion of a node take a bit longer
d) **Implementing** a doubly linked list is easier than singly linked list

25. What is a memory efficient double linked list?
a) **Each** node has only one pointer to traverse the list back and forth
b) The list has breakpoints for faster traversal
c) An auxiliary singly linked list acts as a helper list to traverse through the doubly linked list
d) A doubly linked list that uses bitwise AND operator for storing addresses

26. How do you calculate the pointer difference in a memory efficient double linked list?
    a) head xor tail
    b) **pointer** to previous node xor pointer to next node
    c) pointer to previous node – pointer to next node
    d) pointer to next node – pointer to previous node

27. 6. What is the worst case time complexity of inserting a node in a doubly linked list?
    a) O(nlogn)
    b) O(logn)
    c) O(n)
    d) O(1)

28. What differentiates a circular linked list from a normal linked list?
    a) You cannot have the 'next' pointer point to null in a circular linked list
    b) It is faster to traverse the circular linked list
    c) **You** may or may not have the 'next' pointer point to null in a circular linked list
    d) Head node is known in circular linked list

29. What is the time complexity of searching for an element in a circular linked list?
    a) O(n)
    b) O(nlogn)
    c) O(1)
    d) O(n^2)

30. Which of the following application makes use of a circular linked list?
    a) Undo operation in a text editor
    b) Recursive function calls
    c) Allocating CPU to resources
    d) Implement Hash Tables

31. Which of the following is false about a circular linked list?
    a) Every node has a successor
    b) Time complexity of inserting a new node at the head of the list is O(1)
    c) Time complexity for deleting the last node is O(n)
    d) We can traverse the whole circular linked list by starting from any point

32. Consider a small circular linked list. How to detect the presence of cycles in this list effectively?
    a) Keep one node as head and traverse another temp node till the end to check if its 'next points to head
    b) Have fast and slow pointers with the fast pointer advancing two nodes at a time and slow pointer advancing by one node at a time

c) Cannot determine, you have to pre-define if the list contains cycles
d) Circular linked list itself represents a cycle. So no new cycles cannot be generated

33. Which of the following real world scenarios would you associate with a stack data structure?
a) **piling up** of chairs one above the other
b) people standing in a line to be serviced at a counter
c) offer services based on the priority of the customer
d) tatkal Ticket Booking in IRCTC

34. What does 'stack underflow' refer to?
a) accessing item from an undefined stack
b) adding items to a full stack
c) **removing** items from an empty stack
d) index out of bounds exception

35. What is the time complexity of pop() operation when the stack is implemented using an array?
a) O(1)
b) O(n)
c) O(logn)
d) O(nlogn)

36. 6. Which of the following array position will be occupied by a new element being pushed for a stack of size N elements(capacity of stack > N).
a) S[N-1]
b) S[N]
c) S[1]
d) S[0]

37. What happens when you pop from an empty stack while implementing using the Stack ADT in Java?
a) Undefined error
b) Compiler displays a warning
c) EmptyStackException is thrown
d) NoStackException is thrown

38. Array implementation of Stack is not dynamic, which of the following statements supports this argument?
a) **space** allocation for array is fixed and cannot be changed during run-time
b) user unable to give the input for stack operations
c) a runtime exception halts execution
d) improper program compilation

39. Which of the following array element will return the top-of-the-stack-element for a stack of size N elements(capacity of stack > N).

a) S[N-1]
b) S[N]
c) S[N-2]
d) S[N+1]

40. What is the best case time complexity of deleting a node in Singly Linked list?
a) O (n)
b) O (n^2)
c) O (nlogn)
d) O (1)

41. Which of the following statements are not correct with respect to Singly Linked List(SLL) and Doubly Linked List(DLL)?
a) Complexity of Insertion and Deletion at known position is O(n) in SLL and O(1) in DLL
b) SLL uses lesser memory per node than DLL
c) DLL has more searching power than SLL
d) **Number** of node fields in SLL is more than DLL

42. In linked list implementation of queue, if only front pointer is maintained, which of the following operation take worst case linear time?
a) Insertion
b) Deletion
c) To empty a queue
d) Both Insertion and To empty a queue

43. In linked list implementation of a queue, where does a new element be inserted?
a) At the head of link list
b) At the centre position in the link list
c) At the tail of the link list
d) At any position in the linked list

44. In linked list implementation of a queue, front and rear pointers are tracked. Which of these pointers will change during an insertion into a NONEMPTY queue?
a) Only front pointer
b) Only rear pointer
c) Both front and rear pointer
d) No pointer will be changed

45. In linked list implementation of a queue, front and rear pointers are tracked. Which of these pointers will change during an insertion into EMPTY queue?
a) Only front pointer
b) Only rear pointer

c) Both front and rear pointer
d) No pointer will be changed

46. In linked list implementation of a queue, from where is the item deleted?
a) At the head of link list
b) At the centre position in the link list
c) At the tail of the link list
d) Node before the tail

47. In linked list implementation of a queue, the important condition for a queue to be empty is?
a) FRONT is null
b) REAR is null
c) LINK is empty
d) FRONT==REAR-1

48. The essential condition which is checked before insertion in a linked queue is?
a) Underflow
b) **Overflow**
c) Front value
d) Rear value

49. The essential condition which is checked before deletion in a linked queue is?
a) **Underflow**
b) Overflow
c) Front value
d) Rear value

50. Which of the following is true about linked list implementation of queue?
a) **In** push operation, if new nodes are inserted at the beginning of linked list, then in pop operation, nodes must be removed from end
b) In push operation, if new nodes are inserted at the beginning, then in pop operation, nodes must be removed from the beginning
c) In push operation, if new nodes are inserted at the end, then in pop operation, nodes must be removed from end
d) In push operation, if new nodes are inserted at the end, then in pop operation, nodes must be removed from beginning

51. What is the time complexity of deleting from the rear end of the dequeue implemented with a singly linked list?
a) O(nlogn)
b) O(logn)
c) O(n)
d) O(n^2)

UNIT 4 : Stack

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. The data structure required to check whether an expression contains balanced parenthesis is?
a) Stack
b) Queue
c) Array
d) Tree
Answer: a

7. Which data structure is needed to convert infix notation to postfix notation?
a) Branch
b) Tree
c) Queue
d) Stack
Answer: d

8. Which data structure is used for implementing recursion?
a) Queue
b) Stack
c) Array
d) List
Answer: b

9. Which of the following statement(s) about stack data structure is/are NOT correct?
a) Linked List are used for implementing Stacks
b) Top of the Stack always contain the new node
c) Stack is the FIFO data structure
d) Null link is present in the last node at the bottom of the stack
Answer: c

10. The type of expression in which operator succeeds its operands is?
a) Infix Expression
b) Prefix Expression
c) Postfix Expression
d) Both Prefix and Postfix Expressions
Answer: c

11. If the elements “A”, “B”, “C” and “D” are placed in a stack and are deleted one at a time, what is the order of removal?
a) ABCD
b) DCBA
c) DCAB
d) ABDC
Answer: b

12. What is the result of the following operation
Top (Push (S, X))
a) X
b) Null
c) S
d) None
Answer: a

13. Which data structure is used for implementing recursion?
a) Queue
b) Stack
c) Array
d) List
Answer: b

`14. Consider the linked list implementation of a stack. Which of the following node is considered as Top of the stack?
a) First node
b) Last node
c) Any node
d) Middle node
ANSWER: a

15. Consider the following operation performed on a stack of size 5.
1) Push(1);2) Pop();3) Push(2);4) Push(3);5) Pop();6) Push(4);7) Pop();
8) Pop();9) Push(5);
After the completion of all operation, the no of element present on stack are
a) 1
b) 2
c) 3
d) 4
ANSWER: a

16. Which of the following is not an inherent application of stack?
a) Reversing a string
b) Evaluation of postfix expression
c) Implementation of recursion
d) Job scheduling
ANSWER: d

17. Which of the following operation take worst case linear time in the array implementation of stack?
a) Push
b) Pop
c) IsEmpty
d) None
ANSWER: d

18. The type of expression in which operator Preceeds its operands is?
a) Infix Expression
b) pre fix Expression
c) postfix Expression

d) None
ANSWER: b

`19. Which of the following application generally use a stack?
a) Parenthesis balancing program
b) Syntax analyzer in compiler
c) Keeping track of local variables at run time
d) All of the above
ANSWER: d

20. What is the minimum number of stacks of size n required to implement a queue of size n?
a) One
b) Two
c) Three
d) Four
ANSWER: b

21. What will be the initial value with which top is initialized.
a) 1
b) -1
c) 0
d) Garbage
ANSWER: d

22.Which of the following is true about linked list implementation of stack?

a) In push operation, if new nodes are inserted at the beginning of linked list, then in pop
b) operation, nodes must be removed from end.
In push operation, if new nodes are inserted at the end, then in pop operation, nodes must be removed from the beginning.
c) Both of the above
d) None of the above
Answer:d

23. 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

24. Other names for the insertion and deletion operations in Stacks?
a) PUSH – insertion, POP –Deletion.
b) PUSH – Deletion, POP – Insertion.
c) Both A and B are valid.
d) Only A is true.
Answer: d

25.In stacks, the possibility to do insertion and deletion is ____
a) At one end only
b) at both ends
c) Both A & B.
d) None
Answer: a

26. Pick from the following which represents an element in a stack.
a) SIZE
b) ITEM
c) TOP
d) BOTTOM
Answer: b

27. At which position in the stacks , the operations are being done.
a) TOP
b) SIZE
c) POP
d) PUSH
Answer: a

28. It is impossible to do ___ operation on empty stack.
a) PUSH
b) POP
c) STATUS
d) None
Answer: b

29. From the following which are static and dynamic representations of stacks?
a) Arrays – Static, Linked lists – Dynamic.
b) Linked lists – Static, Arrays – Dynamic.
c) Queues –Dynamic ,Arrays – Static
d) None of the mentioned.
Answer: a

30. Pick out the odd one which is not related to stacks.
a) LIFO list

b) FIFO list
c) Push down lists.
d) Piles
Answer: b

31. Can we delete a node at front of a stack by using POP operation?
a) YES
b) NO
Answer: a

32.The concept "Pile of Plates" is applicable in which one of the following.
a) Stack
b) Queue
c) Linked list
d) Tree
Answer: a

33) In liked representation of stack ……. holds the elements of the stack.
a) INFO fields
b) TOP fields
c) LINK fields
d) NULL fields
Answer: a

34) In the linked representation of the stack ……… behaves as the top pointer variable of stack.
a) Stop pointer
b) Begin pointer
c) Start pointer
d) Avail pointer
Answer: c

35) In linked representation of stack the null pointer of the last node in the list signals ……….
a) Beginning of the stack
b) Bottom of the stack
c) Middle of the stack
d) In between some value
Answer: b

36) What happens when you push a new node onto a stack?
a) The new node is placed at the front of the linked list
b) The new node is placed at the back of the linked list
c) The new node is placed at the middle of the linked list
d) No Changes happens
Answer: a

37) The retrieval of items in a stack is ……….. operation.
a) Push
b) Pop
c) retrieval
d) access
Answer: b

38) Which is the pointer associated with the stack?
a) FIRST
b) FRONT
c) TOP
d) REAR
Answer: c

39) The elements are removal from a stack in ………. order.
a) Reverse
b) Hierarchical
c) Alternative
d) Sequential
Answer: a

40) Which of the following is an application of stack?
a) finding factorial
b) tower of Hanoi
c) infix to postfix
d) all of the above
Answer: d

41. What is the value of the postfix expression 6 3 2 4 + – *?
a) 1
b) 40
c) 74
d) -18
Answer: d

42. 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

d) 4
Answer: d

43. 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

44. The postfix form of A*B+C/D is?
a) *AB/CD+
b) AB*CD/+
c) A*BC+/D
d) ABCD+/*
Answer: b

45. 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

46.The prefix form of an infix expression p + q – r * t is?
+ pq – *rt
a) – +pqr * t
b) – +pq * rt
c) – + * pqrt
Answer: c

47.What is the value of the postfix expression 6 3 2 4 + – *:
Something between -5 and -15
a) Something between 5 and -5
b) Something between 5 and 15
c) Something between 15 and 100
Answer: d

48. The result of evaluating the postfix expression 5, 4, 6, +, *, 4, 9, 3, /, +, * is?
a) 600
b) 350
c) 650
d) 588
Answer: b

49. Which data structure can be used to test a palindrome?

a) Tree
b) Heap
c) Stack
d) Priority queue

Answer: c

50. At the end of last operation, total number of element present in stack are

a) 3
b) 2
c) 1
d) 0

Answer: a

Unit 5: Queue

1. 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

2. The data structure required for Breadth First Traversal on a graph is?
a) Stack
b) Array
c) Queue
d) Tree
Answer: c

3. 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

4. Circular Queue is also known as ________
a) Ring Buffer
b) Square Buffer
c) Rectangle Buffer
d) Curve Buffer
Answer: a

5. 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

6. 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

7. 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

8. 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

9. 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

10. In linked list implementation of a queue, where does a new element be inserted?
a) At the head of link list
b) At the tail of the link list
c) At the centre position in the link list
d) None
ANSWER: b

11. In the array implementation of circular queue, which of the following operation take worst case linear time?
a) Insertion
b) Deletion
c) To empty a queue
d) None
ANSWER: d

12. In linked list implementation of queue, if only front pointer is maintained, which of the following operation take worst case linear time?
a) Insertion
b) Deletion
c) To empty a queue

d) Both a) and c)
ANSWER: d

13. If the MAX_SIZE is the size of the array used in the implementation of circular queue. How is rear manipulated while inserting an element in the queue?
a) rear=(rear%1)+MAX_SIZE
b) rear=rear%(MAX_SIZE+1)
c) rear=(rear+1)%MAX_SIZE
d) rear=rear+(1%MAX_SIZE)
ANSWER: c

14. If the MAX_SIZE is the size of the array used in the implementation of circular queue, array index start with 0, front point to the first element in the queue, and rear point to the last element in the queue. Which of the following condition specify that circular queue is FULL?
a) Front=rear= -1
b) Front=(rear+1)%MAX_SIZE
c) Rear=front+1
d) Rear=(front+1)%MAX_SIZE
ANSWER: b

15. A circular queue is implemented using an array of size 10. The array index starts with 0, front is 6, and rear is 9. The insertion of next element takes place at the array index.
a) 0
b) 7
c) 9
d) 10
ANSWER: a

16. What is the worst case time complexity of a sequence of n queue operations on an initially empty queue?
a) θ (n)
b) θ (n + k)
c) θ (nk)
d) θ (n2)
ANSWER: a

17. If the MAX_SIZE is the size of the array used in the implementation of circular queue, array index start with 0, front point to the first element in the queue, and rear point to the last element in the queue. Which of the following condition specify that circular queue is EMPTY?
a) Front=rear=0
b) Front= rear=-1
c) Front=rear+1
d) Front=(rear+1)%MAX_SIZE
ANSWER: b

18. 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

19. In linked list implementation of a queue, front and rear pointers are tracked. Which of these pointers will change during an insertion into a NONEMPTY queue?
a) Only front pointer
b) Only rear pointer
c) Both front and rear pointer
d) None of the front and rear pointer
ANSWER: b

20. 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

21. In linked list implementation of a queue, front and rear pointers are tracked. Which of these pointers will change during an insertion into EMPTY queue?
a) Only front pointer
b) Only rear pointer
c) Both front and rear pointer
d) None
ANSWER: c

22. An array of size MAX_SIZE is used to implement a circular queue. Front, Rear, and count are tracked. Suppose front is 0 and rear is MAX_SIZE -1. How many elements are present in the queue?
a) Zero
b) One
c) MAX_SIZE-1
d) MAX_SIZE
ANSWER: d

23. Suppose a circular queue of capacity (n-1) elements is implemented with an array of n elements. Assume that the insertion and deletion operations are carried out using REAR and FRONT as array index variables, respectively. Initially REAR=FRONT=0. The conditions to detect queue full and queue is empty are?

a) Full: (REAR+1)mod n == FRONTEmpty: REAR==FRONT

b) Full: (REAR+1)mod n == FRONTEmpty: (FRONT+1) mod n == REAR
c) Full: REAR==FRONTEmpty: (REAR+1) mod n==FRONT
d) Full: (FRONT+1)mod n==REAREmpty: REAR==FRONT
ANSWER: a

24. Which one of the following is an application of Queue Data Structure?
When a resource is shared among multiple consumers.
a) When data is transferred asynchronously (data not necessarily received at same rate as sent) between two processes
b) Load Balancing
c) All of the above
Answer: D

25. In linked list implementation of queue, if only front pointer is maintained, which of the following operation take worst case linear time?
a) Insertion
b) Deletion
c) To empty a queue
d) Both Insertion and To empty a queue
Answer: D

26.How many stacks are needed to implement a queue. Consider the situation where no other data structure like arrays, linked list is available to you.
1
a) 2
b) 3
c) 4
Answer: B

27.In a Queue, if a user tries to remove an element from empty Queue it is called _________.
Underflow
a) Empty collection
b) Overflow
c) Garbage Collection
Answer: A

28. In case of insertion into a linked queue, a node borrowed from the __________ list is inserted in the queue.
REAR
a) None of the mentioned
b) AVAIL
c) FRONT

Answer: A

29.If the MAX_SIZE is the size of the array used in the implementation of circular queue. How is rear manipulated while inserting an element in the queue?
rear=(rear%1)+MAX_SIZE
a) rear=rear%(MAX_SIZE+1)
b) rear=(rear+1)%MAX_SIZE
c) rear=rear+(1%MAX_SIZE)
View Answer

30. Queues serve major role in
Simulation of recursion
a) Simulation of arbitrary linked list
b) Simulation of limited resource allocation
c) All of the mentioned
Answer: C

31. Which of the following is not the type of queue?
Ordinary queu
a) Single ended queue
b) Circular queue
c) Priority queue
Answer: B

32.What kind of a datastructure does a queue is?

a) Linear
b) B non-linear
c) both A&B
d) none
Answer: A

33.Which one of the following terms we use to mention the number of elements that a queue can hold?
a) LENGTH
b) SIZE
c) CAPACITY
d) DATA
Answer: A

34. Similarly in DEQUEUEs, insertion is performed at ___ end whereas the deletion is performed at __ end.

a) FRONT, REAR
b) REAR, FRONT.
c) Both A & B
d) None of the above
Answer: C

35.Consider P,Q,R and S are the four elements in a queue. If we delete an element at a time then on which order they will get deleted?
a) PQRS
b) SRQP
c) PSQR
d) SRQP
Answer: A

36.A circular queue is implemented using an array of size 10. The array index starts with 0, front is 6, and rear is 9. The insertion of next element takes place at the array index of__.
a) 0
b) 7
c) 9
d) 10
Answer: A

37.In linked list implementation of a queue, front and rear pointers are tracked. Which of these pointers will change during an insertion into a NONEMPTY queue?
a) Only front pointer
b) Only rear pointer
c) Both front and rear pointer
d) None of the front and rear pointer
Answer: B

38.The implementation of Radix sort can be done with the help of ____.
a) Linked list
b) Stack
c) Queue
d) Possible with all the above.
Answer: C

39.Which one of the following is an application(s) of Queue Data Structure?
a) When a resource is shared among multiple consumers
b) When data is transferred asynchronously between the two processes.
c) Load balancing
d) all the above
Answer: D

40. How many stacks are needed to implement a queue?
a) 1
b) 2
c) 3
d) 4
Answer: B

41. In Queue, ENQUEUE means____ whereas DEQUEUE refers____.
a) an insertion operation, a deletion operation.
b) End of the queue, defining a queue.
c) Both A and B.
d) None of the above are true.
Answer: A

42. ___________of the queue added a new nodes
a) Front
b) middle
c) back
d) Both A and B
e) None
Answer: c

43. Major role of queue server in ______________
a) Simulation of heapsort
b) Simulation of arbitrary linked list
c) Simulation of limited resource allocation
d) Simulation of recursion
e) Both a&b
Answer: c

44. Which elements not in middle but can be inserted or deleted at/from both the ends?
a) Circular queue
b) Priority queue
c) Queue
d) DE queue
e) All of these
Answer: d

45. Circular Queue is also called ________
a) Square Buffer
b) Ring Buffer
c) Rectangle Buffer
d) Curve Buffer
e) None of these
Answer: b

46. For Breadth-First Traversal on a graph is the data structure required?
a) Stack
b) queue
c) array
d) Tree
e) Both a&b
Answer: b

47. Which deletion can be insertion take place only at the other end(rear) and done from one end (front)?
a) linked list
b) Stack
c) Tree
d) queue
e)both a&c
Answer: d

48. In what order will they be removed If the elements “A”, “B”, “C” and “D” are placed in a queue and are deleted one at a time
a) ABCD
b) DCAB
c) DCBA
d) ABDC
e) All of the above
Answer: a

49. if implemented using an array of size MAX_SIZE, gets full when
a) Front = (rear + 1)mod MAX_SIZE
b) Front = rear + 1
c) Rear = MAX_SIZE – 1
d) Rear = front
e) None of above
Answer: c

50. Which is not the type of queue?
a) Single ended queue
b) Ordinary queue
c) Circular queue

d) Priority queue
e) Both c&d
Answer: a

Unit 6 : Tree

1. How many children does a binary tree have?
a) 2
b) any number of children
c) 0 or 1 or 2
d) 0 or 1

2. What is/are the disadvantages of implementing tree using normal arrays?
a) difficulty in knowing children nodes of a node
b) difficult in finding the parent of a node
c) have to know the maximum number of nodes possible before creation of trees
d) difficult to implement

3. What must be the ideal size of array if the height of tree is 'l'?
a) 2^l-1
b) l-1
c) l
d) 2l

4. What are the children for node 'w' of a complete-binary tree in an array representation?
a) 2w and 2w+1
b) 2+w and 2-w
c) w+1/2 and w/2
d) w-1/2 and w+1/2

5. What is the parent for a node 'w' of a complete binary tree in an array representation when w is not 0?
a) floor(w-1/2)
b) ceil(w-1/2)
c) w-1/2
d) w/2

6. If the tree is not a complete binary tree then what changes can be made for easy access of children of a node in the array?
a) every node stores data saying which of its children exist in the array
b) no need of any changes continue with 2w and 2w+1, if node is at i
c) keep a seperate table telling children of a node
d) use another array parallel to the array with tree

7. The number of edges from the root to the node is called ___________ of the tree.
a) Height
b) Depth

c) Length
d) Width

8. The number of edges from the node to the deepest leaf is called _________ of the tree.
a) Height
b) Depth
c) Length
d) Width

9. 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

10. 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

11. 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)

12. 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

13. 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

14. 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

15. 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

16. 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))

17. 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

18. 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

19. 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)

20. 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

21. What will be the height of a balanced full binary tree with 8 leaves?
a) 8
b) 5

c) 6
d) 4

22. 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

23. 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

24. 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

25. 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)

26. 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

27. 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)

28. 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

29. The no of external nodes in a full binary tree with n internal nodes is?
a) A. n
b) B. n+1
c) C. 2n
d) D. 2n + 1

30. Which of the following is a true about Binary Trees?
a) Every binary tree is either complete or full.
b) Every complete binary tree is also a full binary tree.
c) C. Every full binary tree is also a complete binary tree.
d) None of the above

31. A Binary Tree can have
a) Can have 2 children
b) Can have 1 children
c) Can have 0 children
d) All of the above

32.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

33.The difference between the external path length and the internal path length of a binary tree with n internal nodes is?
a) 1
b) n
c) n + 1
d) 2n

34. Which of the following statements is false ?
a) Every tree is a bipartite graph
b) A tree contains a cycle
c) A tree with n nodes contains n-1 edges
d) A tree is a connected graph

35. Which of the following remarks about Tree- indexing are true?
a) It is an m-ary tree

b) Successful searches should terminate in leaf nodes
c) Unsuccessful searches may terminate in leaf nodes level of the tree structure
d) All of these

36. A binary tree in which if all its levels except possibly the last, have the maximum number of nodes and all the nodes at the last level appear as far left as possible, is called
a) Full binary tree
b) Binary Search Tree
c) Threaded tree
d) Complete binary tree

37. A complete binary tree with the property that the value at each node is at least as large as the values at its children is called
a) binary search tree
b) Binary Tree
c) Completely balanced tree
d) Heap

38. A full binary tree with n leaves contains
a) n nodes
b) log2n nodes
c) 2n - 1 nodes
d) 2n+1 nodes

39. Which of the following is a true about Binary Trees
a) Every binary tree is either complete or full.
b) Every complete binary tree is also a full binary tree.
c) Every full binary tree is also a complete binary tree.
d) No binary tree is both complete and full.
(E) None of the above

40. If arity of operators is fixed, then which of the following notations can be used to parse expressions without parentheses?

a) Infix Notation (Inorder traversal of a expression tree)
b) Postfix Notation (Postorder traversal of a expression tree)
c) Prefix Notation (Preorder traversal of a expression tree)

a) b and c
b) Only b
c) a, b and c
d) None of them

41. n a complete k-ary tree, every internal node has exactly k children or no child. The number of leaves in such a tree with n internal nodes is:

a) nk

b) (n – 1) k+ 1

c) n(k – 1) + 1

d) n(k – 1)

42. What are the main applications of tree data structure?
1) Manipulate hierarchical data
2) Make information easy to search (see tree traversal).
3) Manipulate sorted lists of data
4) Router algorithms
5) Form of a multi-stage decision-making, like Chess Game.
6) As a workflow for compositing digital images for visual effects
a) 1, 2, 3, 4 and 6
b) 1, 2, 3, 4 and 5
c) 1, 3, 4, 5 and 6
d) 1, 2, 3, 4, 5 and 6

43. The maximum number of binary trees that can be formed with three unlabeled nodes is:
a) 1
b) 5
c) 4
d) 3

44. What is the worst case time complexity for search, insert and delete operations in a general Binary Search Tree?
a) O(n) for all
b) O(Logn) for all
c) O(Logn) for search and insert, and O(n) for delete
d) O(Logn) for search, and O(n) for insert and delete

45. We are given a set of n distinct elements and an unlabeled binary tree with n nodes. In how many ways can we populate the tree with the given set so that it becomes a binary search tree?
a) 0
b) 1
c) n!
d) (1/(n+1)).2nCn

46. If arity of operators is fixed, then which of the following notations can be used to parse expressions without parentheses?

a) Infix Notation (Inorder traversal of a expression tree)
b) Postfix Notation (Postorder traversal of a expression tree)
c) Prefix Notation (Preorder traversal of a expression tree)

a) b and c
b) Only b
c) a, b and c
d) None of them

47. The maximum number of binary trees that can be formed with three unlabeled nodes is:
a) 1
b) 5
c) 4
d) 3

48. In delete operation of BST, we need inorder successor (or predecessor) of a node when the node to be deleted has both left and right child as non-empty. Which of the following is true about inorder successor needed in delete operation?
a) Inorder Successor is always a leaf node
b) Inorder successor is always either a leaf node or a node with empty left child
c) Inorder successor may be an ancestor of the node
d) Inorder successor is always either a leaf node or a node with empty right child

49. Suppose the numbers 7, 5, 1, 8, 3, 6, 0, 9, 4, 2 are inserted in that order into an initially empty binary search tree. The binary search tree uses the usual ordering on natural numbers. What is the in-order traversal sequence of the resultant tree?
a) 7 5 1 0 3 2 4 6 8 9
b) 0 2 4 3 1 6 5 9 8 7
c) 0 1 2 3 4 5 6 7 8 9
d) 9 8 6 4 2 3 0 1 5 7

50. The following numbers are inserted into an empty binary search tree in the given order: 10, 1, 3, 5, 15, 12, 16. What is the height of the binary search tree (the height is the maximum distance of a leaf node from the root)? (GATE CS 2004)
a) 2
b) 3
c) 4
d) 6

Unit 7 : Graph

1. Which of the following statements for a simple graph is correct?
   a) Every path is a trail
   b) Every trail is a path
   c) Every trail is a path as well as every path is a trail
   d) Path and trail have no relation
2. What is the number of edges present in a complete graph having n vertices?
   a) (n*(n+1))/2
   b) (n*(n-1))/2
   c) n
   d) Information given is insufficient
3. In a simple graph, the number of edges is equal to twice the sum of the degrees of the vertices.
   a) True
   b) False

4. A connected planar graph having 6 vertices, 7 edges contains _____________ regions.
   a) 15
   b) 3
   c) 1
   d) 11

5. If a simple graph G, contains n vertices and m edges, the number of edges in the Graph G'(Complement of G) is ___________
   a) (n*n-n-2*m)/2
   b) (n*n+n+2*m)/2
   c) (n*n-n-2*m)/2
   d) (n*n-n+2*m)/2

6. Which of the following properties does a simple graph not hold?
   a) Must be connected
   b) Must be unweighted
   c) Must have no loops or multiple edges
   d) Must have no multiple edges

7. What is the maximum number of edges in a bipartite graph having 10 vertices?
   a) 24
   b) 21
   c) 25
   d) 16

8. Which of the following is true?
   a) A graph may contain no edges and many vertices

b) A graph may contain many edges and no vertices
c) A graph may contain no edges and no vertices
d) A graph may contain no vertices and many edges

9. For a given graph G having v vertices and e edges which is connected and has no cycles, which of the following statements is true?
a) v=e
b) v = e+1
c) v + 1 = e
d) v = e-1

10. For which of the following combinations of the degrees of vertices would the connected graph be eulerian?
a) 1,2,3
b) 2,3,4
c) 2,4,5
d) 1,3,5

11. A graph with all vertices having equal degree is known as a __________
a) Multi Graph
b) Regular Graph
c) Simple Graph
d) Complete Graph

12. Which of the following ways can be used to represent a graph?
a) Adjacency List and Adjacency Matrix
b) Incidence Matrix
c) Adjacency List, Adjacency Matrix as well as Incidence Matrix
d) No way to represent

13. What is the number of edges present in a complete graph having n vertices?
a) (n*(n+1))/2
b) (n*(n-1))/2
c) n
d) Information given is insufficient

14. If a simple graph G, contains n vertices and m edges, the number of edges in the Graph G'(Complement of G) is ___________
a) (n*n-n-2*m)/2
b) (n*n+n+2*m)/2
c) (n*n-n-2*m)/2
d) (n*n-n+2*m)/2

15. Which of the following properties does a simple graph not hold?
a) Must be connected
b) Must be unweighted
c) Must have no loops or multiple edges
d) All of the mentioned

16. Which of the following is true?
a) A graph may contain no edges and many vertices
b) A graph may contain many edges and no vertices
c) A graph may contain no edges and no vertices
d) None of the mentioned

17. For a given graph G having v vertices and e edges which is connected and has no cycles, which of the following statements is true?
a) v=e
b) v = e+1
c) v + 1 = e
d) None of the mentioned

18. for which of the following combinations of the degrees of vertices would the connected graph be eulerian?
a) 1,2,3
b) 2,3,4
c) 2,4,5
d) 1,3,5

19. A graph with all vertices having equal degree is known as a __________
a) Multi Graph
b) Regular Graph
c) Simple Graph
d) Complete Graph

20. Which of the following ways can be used to represent a graph?
a) Adjacency List and Adjacency Matrix
b) Incidence Matrix
c) Adjacency List, Adjacency Matrix as well as Incidence Matrix
d) None of the mentioned

21. The time complexity to calculate the number of edges in a graph whose information in stored in form of an adjacency matrix is ____________
a) O(V)
b) O(E2)
c) O(E)
d) O(V2)

22. Which of the following statements for a simple graph is correct?
a) Every path is a trail
b) Every trail is a path
c) Every trail is a path as well as every path is a trail
d) None of the mentioned

23. What is the maximum number of possible non zero values in an adjacency matrix of a simple graph with n vertices?
a) (n*(n-1))/2
b) (n*(n+1))/2
c) n*(n-1)
d) n*(n+1)

24. Which of the following is an advantage of adjacency list representation over adjacency matrix representation of a graph?
a) In adjacency list representation, space is saved for sparse graphs.
b) DFS and BSF can be done in O(V + E) time for adjacency list representation. These operations take O(V^2) time in adjacency matrix representation. Here is V and E are number of vertices and edges respectively.
c) Adding a vertex in adjacency list representation is easier than adjacency matrix representation.
d) All of the above

25. A connected planar graph having 6 vertices, 7 edges contains _____________ regions.
a) 15
b) 3
c) 1
d) 11

26. On which of the following statements does the time complexity of checking if an edge exists between two particular vertices is not, depends?
a) Depends on the number of edges
b) Depends on the number of vertices
c) Is independent of both the number of edges and vertices
d) It depends on both the number of edges and vertices

27. The degree sequence of a simple graph is the sequence of the degrees of the nodes in the graph in decreasing order. Which of the following sequences can not be the degree sequence of any graph?
I. 7, 6, 5, 4, 4, 3, 2, 1
II. 6, 6, 6, 6, 3, 3, 2, 2
III. 7, 6, 6, 4, 4, 3, 2, 2
IV. 8, 7, 7, 6, 4, 2, 1, 1

a) I and II
b) III and IV
c) IV only
d) II and IV

28. What is the maximum number of edges in a bipartite graph having 10 vertices?
a) 24
b) 21
c) 25
d) 16

29. Possible number of labelled simple Directed, Pseudo and Multigarphs exist having 2 vertices?
a) 3, Infinite, 4
b) 4, 3, Infinite
c) 4, Infinite, infinite
d) 4, Infinite, Infinite

30. The most efficient algorithm for finding the number of connected components in an undirected graph on n vertices and m edges has time complexity.
(a) \theta(n)
(b) \theta(m)
(c) \theta(m + n)
d) \theta(mn)

31. What is time complexity to check if a string(length S1) is a substring of another string(length S2) stored in a Directed Acyclic Word Graph, given S2 is greater than S1?
a) O(S1)
b) O(S2)
c) O(S1+S2)
d) O(1)

32. Which of the following is an advantage of adjacency list representation over adjacency matrix representation of a graph?
a) In adjacency list representation, space is saved for sparse graphs.
b) DFS and BSF can be done in O(V + E) time for adjacency list representation. These operations take O(V^2) time in adjacency matrix representation. Here is V and E are number of vertices and edges respectively.
c) Adding a vertex in adjacency list representation is easier than adjacency matrix representation.
d) All of the above

33. The time complexity of computing the transitive closure of a binary relation on a set of n elements is known to be:
a) O(n)

b) O(nLogn)
c) O(n ^ (3/2))
d) O(n^3)

34. The most efficient algorithm for finding the number of connected components in an undirected graph on n vertices and m edges has time complexity.

a) (n)

b) (m)

c) (m + n)

d) (mn)

35. Consider an undirected unweighted graph G. Let a breadth-first traversal of G be done starting from a node r. Let d(r, u) and d(r, v) be the lengths of the shortest paths from r to u and v respectively, in G. lf u is visited before v during the breadth-first traversal, which of the following statements is correct? (GATE CS 2001)
a) d(r, u) < d (r, v)
b) d(r, u) > d(r, v)
c) d(r, u) <= d (r, v)
d) None of the above

36. How many undirected graphs (not necessarily connected) can be constructed out of a given set V= {V 1, V 2,…V n} of n vertices ?
a) n(n-l)/2
b) 2^n
c) n!
d) 2^(n(n-1)/2)

37. Which of the following statements is/are TRUE for an undirected graph?
P: Number of odd degree vertices is even
Q: Sum of degrees of all vertices is even
a) P Only
b) Q Only
c) Both P and Q
d) Neither P nor Q

38. Consider an undirected random graph of eight vertices. The probability that there is an edge between a pair of vertices is 1/2. What is the expected number of unordered cycles of length three?
a) 1/8
b) 1
c) 7
d) 8

39. Given an undirected graph G with V vertices and E edges, the sum of the degrees of all vertices is
a) E
b) 2E
c) V
d) 2V

40. What is the maximum number of possible non zero values in an adjacency matrix of a simple graph with n vertices?
a) (n*(n-1))/2
a) (n*(n+1))/2
c) n*(n-1)
d) n*(n+1)

41. On which of the following statements does the time complexity of checking if an edge exists between two particular vertices is not, depends?
a) Depends on the number of edges
a) Depends on the number of vertices
c) Is independent of both the number of edges and vertices
d) It depends on both the number of edges and vertices

42. Assuming value of every weight to be greater than 10, in which of the following cases the shortest path of a directed weighted graph from 2 vertices u and v will never change?
a) add all values by 10
a) subtract 10 from all the values
c) multiply all values by 10
d) In both the cases of multiplying and adding by 10

43. A connected planar graph having 6 vertices, 7 edges contains _____________ regions.
a) 15
b) 3
c) 1
d) 11

44. If a connected Graph (G) contains n vertices what would be the rank of its incidence matrix?
a) n-1
b) values greater than n are possible
c) values less than n-1 are possible
d) Insufficient Information is given

45. Possible number of labeled simple Directed, Pseudo and Multigarphs exist having 2 vertices?
a) 3, Infinite, 4
b) 4, 3, Infinite
c) 4, Infinite, infinite

d) 4, Infinite, Infinite

46. What is the number of vertices of degree 2 in a path graph having n vertices,here n>2.
a) n-2
b) n
c) 2
d) 0

47. What is time complexity to check if a string(length S1) is a substring of another string(length S2) stored in a Directed Acyclic Word Graph, given S2 is greater than S1?
a) O(S1)
b) O(S2)
c) O(S1+S2)
d) O(1)

48. For an undirected graph G with n vertices and e edges, the sum of the degrees of each vertex is
a) ne
b) 2n
c) 2e
d) e^n

49. Graph traversal is different from a tree traversal, because:
a) trees are not connected
b) graphs may have loops
c) trees have root
d) None of these

50. Graphs are represented using ....
a) Adjacency tree
b) Adjacency linked list
c) Adjacency graph
d) Adjacency queue