500+ Abstract Data Types MCQs with FREE PDF

1. 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: Both (b) and (c)

2. An algorithm is made up of 2 modules M1&M2. If order of M1 is f(n) & M2 is g(n) then the order of algorithm is?

(A) max (f(n),g(n))

(B) min (f(n),g(n))

(C) f(n) + g(n)

(D) f(n) X g(n)

Answer: min (f(n),g(n))

3. 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

Answer: Container of objects of similar types

4. How do you initialize an array in C?

(A) int arr[3] = (1,2,3);

(B) int arr(3) = {1,2,3};

(C) int arr[3] = {1,2,3};

(D) int arr(3) = (1,2,3);

Answer: int arr[3] = {1,2,3};

5. How do you instantiate an array in Java?

(A) int arr[] = new int(3);

(B) int arr[];

(C) int arr[] = new int[3];

(D) int arr() = new int(3);

Answer: int arr[] = new int[3];

6. Which of the following is the correct way to declare a multidimensional array in Java?

(A) int[] arr;

(B) int arr[[]];

(C) int[][]arr;

(D) int[[]] arr;

Answer: int arr[[]];

7. In general, the index of the first element in an array is ___________

(A) 0

(B) -1

(C) 2

(D) 1

Answer: 0

8. Elements in an array are accessed _____________

(A) randomly

(B) sequentially

(C) exponentially

(D) logarithmically

Answer: randomly

9. When does the ArrayIndexOutOfBoundsException occur?

(A) Compile-time

(B) Run-time

(C) Not an error

(D) Not an exception at all

Answer: Run-time

10. What will be the minimum number of jumps required to reach the end of the array arr[] = {1,2,0,0,3,6,8,5}?

a) 1

b) 2

c) 3

d) not possible to reach the end

Answer: not possible to reach the end

11. It is not possible to find the minimum number of steps to reach the end of an array in linear time.

a) true

b) false

Answer: false

12. In how many different ways we can reach the end of the array arr[]={1,3,5,8,9}?

a) 1

b) 2

c) 3

d) 4

Answer: 4

13. It is not possible to reach the end of an array if starting element of the array is 0.

a) true

b) false

Answer: true

14. What is the minimum possible time complexity to find the number of steps to reach the end of an array?

a) O(n)

b) O(n2)

c) O(n3/2)

d) O (1)

Answer: O(n)

15. What will be the minimum number of jumps required to reach the end of the array arr[] = {1,3,6,3,6,8,5}?

a) 1

b) 2

c) 3

d) not possible to reach the end

Answer: 3

[100+ Array and Array Operations MCQs with FREE PDF]

16. Consider an implementation of unsorted singly linked list. Suppose it has its representation with a head and tail pointer. 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

Answer: I, II and III

17. 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

Answer: I and III

18. Consider an implementation of unsorted doubly linked list. Suppose it has its representation with a head pointer and tail pointer. 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 end node of the linked list

(A) I and II

(B) I and III

(C) I, II and III

(D) I, II, III and IV

Answer: I, II, III and IV

19. Consider an implementation of unsorted doubly 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 end node of the linked list

(A) I and II

(B) I and III

(C) I, II and III

(D) I, II, III and IV

Answer: I and III

20. Consider an implementation of unsorted circular 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 end node of the linked list

(A) I and II

(B) I and III

(C) I, II, III and IV

(D) None

Answer: None

21. Consider an implementation of unsorted circular doubly 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 end node of the linked list

(A) I and II

(B) I and III

(C) I, II and III

(D) I, II, III and IV

Answer: I, II, III and IV

22. 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

Answer: Pointer to node

23. 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)

Answer: θ (n)

24. What would be the asymptotic time complexity to add an element in the linked list?

(A) O (1)

(B) O (n)

(C) O (n2)

(D) None

Answer: O (n)

25. What would be the asymptotic time complexity to find an element in the linked list?

(A) O (1)

(B) O (n)

(C) O (n2)

(D) None

Answer: O (n)

26. 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: To avoid collision in hash tables

27. After performing these set of operations, what does the final list look contain?

InsertFront(10);

InsertFront(20);

InsertRear(30);

DeleteFront();

InsertRear(40);

InsertRear(10);

DeleteRear();

InsertRear(15);

display();

a) 10 30 10 15

b) 20 30 40 15

c) 20 30 40 10

d) 10 30 40 15

Answer: 10 30 40 15

28. 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(n2)

Answer: O(n)

29. 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

Answer: Each node has only one pointer to traverse the list back and forth

30. 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

Answer: Each node has only one pointer to traverse the list back and forth

31. 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

Answer: pointer to previous node xor pointer to next node

100+ Doubly Linked List Questions And Answers – Data Structure MCQs

32. 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: O(n)

33. 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: Interrupt handling

34. Which of the following is not an advantage of a 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: Easy to delete elements in any case

35. 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: O(n)

100+ Priority Queue Questions –Data Structure MCQs

36. A queue is a?

a) FIFO (First In First Out) list

b) LIFO (Last In First Out) list.

c) Ordered array

d) Linear tree

Answer: FIFO (First In First Out) list

37. In Breadth First Search of Graph, which of the following data structure is used?

a) Stack

b) Queue

c) Linked list

d) None

Answer: Queue

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) ABCD

Answer: ABCD

39. 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: At the tail of the link list

40. 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: None

41. 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: Both a) and c)

42. 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: rear=(rear+1)%MAX_SIZE

100+ Queue Operations Data Structure MCQs and Answers

43. 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: (rear+1) % CAPACITY

44. 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: overflow

45. What is the time complexity of enqueue operation?

a) O(logn)

b) O(nlogn)

c) O(n)

d) O(1)

Answer: O(1)

46. 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: effective usage of memory

100+ Queue Using Array Program MCQs and Answers

47. 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

Answer: At the tail of the link list

48. 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

Answer: Only rear pointer

49. 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

Answer: Both front and rear pointer

50. In case of insertion into a linked queue, a node borrowed from the __________ list is inserted in the queue.

a) AVAIL

b) FRONT

c) REAR

d) NULL

Answer: AVAIL

51. 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

Answer: At the head of link list

52. 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

Answer: FRONT is null

53. The essential condition which is checked before insertion in a linked queue is?

a) Underflow

b) Overflow

c) Front value

d) Rear value

Answer: Overflow

100+ Queue Linked List Question and Answer – Data Structure MCQs

54. Consider you have a stack whose elements in it are as follows.

5 4 3 2 << top
Where the top element is 2.
You need to get the following stack
6 5 4 3 2 << top

The operations that needed to be performed are (You can perform only push and pop):

a) Push(pop()), push(6), push(pop())

b) Push(pop()), push(6)

c) Push(pop()), push(pop()), push(6)

d) Push(6)

Answer: Push(pop()), push(6), push(pop())

55. A double-ended queue supports operations like adding and removing items from both the sides of the queue. They support four operations like addFront (adding item to top of the queue), addRear (adding item to the bottom of the queue), removeFront (removing item from the top of the queue) and removeRear (removing item from the bottom of the queue). You are given only stacks to implement this data structure. You can implement only push and pop operations. What's the time complexity of performing addFront and addRear? (Assume 'm' to be the size of the stack and 'n' to be the number of elements)

a) O(m) and O(n)

b) O(1) and O(n)

c) O(n) and O(1)

d) O(n) and O(m)

Answer: O(1) and O(n)

56. Why is implementation of stack operations on queues not feasible for a large dataset (Asssume the number of elements in the stack to be n)?

a) Because of its time complexity O(n)

b) Because of its time complexity O(log(n))

c) Extra memory is not required

d) There are no problems

Answer: Because of its time complexity O(n)

57. Consider yourself to be in a planet where the computational power of chips to be slow. You have an array of size 10. You want to perform enqueue some element into this array. But you can perform only push and pop operations. Push and pop operation both take 1 sec respectively. The total time required to perform enQueue operation is?

a) 20

b) 40

c) 42

d) 43

Answer: 43

58. You have two jars, one jar which has 10 rings and the other has none. They are placed one above the other. You want to remove the last ring in the jar. And the second jar is weak and cannot be used to store rings for a long time.

a) Empty the first jar by removing it one by one from the first jar and placing it into the second jar

b) Empty the first jar by removing it one by one from the first jar and placing it into the second jar and empty the second jar by placing all the rings into the first jar one by one

c) There exists no possible way to do this

d) Break the jar and remove the last one

Answer: Empty the first jar by removing it one by one from the first jar and placing it into the second jar and empty the second jar by placing all the rings into the first jar one by one

59. Given only a single array of size 10 and no other memory is available. Which of the following operation is not feasible to implement (Given only push and pop operation)?

a) Push

b) Pop

c) Enqueue

d) Returntop

Answer: Enqueue

60. Given an array of size n, let's assume an element is 'touched' if and only if some operation is performed on it (for example, for performing a pop operation the top element is 'touched'). Now you need to perform Dequeue operation. Each element in the array is touched atleast?

a) Once

b) Twice

c) Thrice

d) Four times

Answer: Four times

100+ Queue using Stacks MCQs and Answers

61. 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)

Answer: Either O(1) or O(n)

62. 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(n2)

Answer: O(n)

63. 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)

Answer: O(1)

64. Which of these is not an application of a linked list?

a) To implement file systems

b) For separate chaining in hash-tables

c) To implement non-binary trees

d) Random Access of elements

Answer: Random Access of elements

100+ Singly Linked List MCQ Questions and Answers

65. 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)

Answer: θ(n)

66. 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(n2)

d) O(n3)

Answer: O(1)

67. What would be the asymptotic time complexity to find an element in the linked list?

a) O(1)

b) O(n)

c) O(n2)

d) O(n4)

Answer: O(n)

68. 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(n2)

d) O(n3)

Answer: O(1)

69. The concatenation of two lists can be performed in O(1) time. Which of the following variation of the linked list can be used?

a) Singly linked list

b) Doubly linked list

c) Circular doubly linked list

d) Array implementation of list

Answer: Circular doubly linked list

100+ Singly Linked Lists Operations Questions and Answers

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

a) Underflow

b) Empty collection

c) Overflow

d) Garbage Collection

Answer: Underflow

71. 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: Overflow

72. 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: There is a Sequential entry that is one by one

73. 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: Data Transfer between two asynchronous process

74. 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: (()(())(()))?

a) 1

b) 2

c) 3

d) 4 or more

Answer: 3

100+Stack Operations Questions and Answers

75. 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)

Answer: O(1)

76. 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]

Answer: S[N]

77. 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

Answer: EmptyStackException is thrown

78. 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

Answer: space allocation for array is fixed and cannot be changed during run-time

79. What does 'stack overflow' 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

Answer: adding items to a full stack

80. Which of the following data structures can be used for parentheses matching?

a) n-ary tree

b) queue

c) priority queue

d) stack

Answer: stack

81. Minimum number of queues to implement stack is ___________

a) 3

b) 4

c) 1

d) 2

Answer: 1

82. The term Push and Pop is related to

a) Queue

b) Stack

c) Both

d) None

Answer: Stack

83. In which data structure element is inserted at one end called Rear and deleted at other end called Front.

a) Stack

b) Queue

c) Both

d) Binary Tree

Answer: Queue

84. To implement a stack using queue(with only enqueue and dequeue operations), how many queues will you need?

a) 1

b) 2

c) 3

d) 4

Answer: 2

100+ Stack using Queues Questions and Answers

1000+ Data Structure MCQs

Application of Stacks
Arrays Types
Types of Lists
Binary Trees
B-Trees
Trees
Heap
Trie
Hash Tables
Graph