BPSC TRE4.0/STET बिहार शिक्षक भर्ती

Computer Science

Data Structure

Array and Linked List MCQ

1. Which of these best describes an array?
   a) A data structure that shows a hierarchical behavior
   b) Container of objects of similar types
   c) Arrays are immutable once initialized
   d) More than one of the above
   e) None of the above

2. 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) More than one of the above

e) None of the above

3. How do you instantiate an array in Java?

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

b) int arr[];

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

d) More than one of the above

e) None of the above

4. 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) More than one of the above

e) None of the above

5. What is the output of the following Java code?

```
public class array
{
    public static void main(String args[])
    {
        int []arr = {1,2,3,4,5};
        System.out.println(arr[2]);
        System.out.println(arr[4]);    }    }
```

a) 3 and 5
b) 5 and 3
c) 2 and 4
d) More than one of the above
e) None of the above

6. What is the output of the following Java code?

```
public class array
{   public static void main(String args[])
    {
        int []arr = {1,2,3,4,5};
        System.out.println(arr[5]);
    }
}
```

a) 4
b) 5
c) ArrayIndexOutOfBoundsException
d) More than one of the above
e) None of the above

7. When does the ArrayIndexOutOfBoundsException occur?

a) Compile-time

b) Run-time

c) Not an error

d) More than one of the above

e) None of the above

8. Which of the following concepts make extensive use of arrays?

a) Binary trees

b) Scheduling of processes

c) Caching

d) Spatial locality

e) More than one of the above

9. What are the advantages of arrays?

a) Objects of mixed data types can be stored

b) Elements in an array cannot be sorted

c) Index of the first element of an array is 1

d) Easier to store elements of the same data type

e) More than one of the above

10. What are the disadvantages of arrays?

a) Data structures like queue or stack cannot be implemented

b) There are chances of wastage of memory space if elements inserted in an array are lesser than the allocated size

c) Index value of an array can be negative

d) Elements are sequentially accessed

e) More than one of the above

11. Assuming int is of 4 bytes, what is the size of int arr[15];?

a) 45

b) 19

c) 30

d) 60

e) None of the above

12. In general, the index of the first element in an array is _____.

a) 0

b) -1

c) 2

d) More than one of the above

e) None of the above

13. Elements in an array are accessed _____.

a) Randomly

b) Sequentially

c) Exponentially

d) Logarithmically

e) More than one of the above

14. Which of the following is the limitation of the array?

a) Elements can be accessed from anywhere

b) The size of the array is fixed

c) Indexing is started from Zero

d) Memory waste if an array's elements are smaller than the size allotted to them

e) More than one of the above

15. In an array int arr[3]={1,2,3}, what will happen if we try to access arr[4] in C/C++?

a) Run-time error

b) 3

c) 0

d) Garbage Value

e) More than one of the above

16. What is the time complexity to insert a single element in the array at the end?

a) O(1)

b) O(n)

c) O(log n)

d) More than one of the above

e) None of the above

17. If we declare a 2D array like int arr[3][4], then it will be stored in the format of _____.

a) 4 rows 3 columns

b) 4 rows 4 columns

c) 3 rows 3 columns

d) 3 rows 4 columns

e) None of the above

18. What is the time complexity for traversing a 2D array?

a) O(n)

b) O(nlogn)

c) $O(n^2)$

d) More than one of the above

e) None of the above

19. What is the output of the following program?

```
#include <bits/stdc++.h>
using namespace std;
int main()
{    int arr[] = { 1, 2, 3, 4, 5 };
    for (int i = 0; i < 5; i++)
        cout << *(arr + i) << endl;
    return 0;}
```

a) 1, 3, 5, 7, 9

b) 1200, 1201, 1202, 1203, 1204

c) 1, 2, 3, 4, 5

d) More than one of the above

e) None of the above

20. What is the output of the following program?

```
#include <iostream>
using namespace std;
int main()
{   int val = 10;
    int* ptr = &val;
    cout << *ptr << endl;
    cout << ptr << endl;
    return 0; }
```

a) Address of val, Address of val

b) Address of val, 10

c) 10, 10

d) More than one of the above

e) None of the above

21. What is the time complexity for inserting/deleting at the beginning of the array?

a) O(1)

b) O(logN)

c) O(NlogN)

d) O(N)

e) None of the above

22. In C++, we can create a dynamic array using the _____ keyword.

a) array
b) this
c) super
d) new
e) None of the above

23. What is the advantage of a dynamic array over a static array?

a) The dynamic array is fixed

b) The dynamic array takes O(n) time

c) The size of the dynamic array is not fixed

d) More than one of the above

e) None of the above

24. How can we create a dynamic array in C?

a) vector()

b) dynamic_array()

c) malloc()

d) More than one of the above

e) None of the above

25. Consider a 2-dimensional array a[4][5], the array is stored in row-major format. If the first element x[0][0] occupies the memory location with address 1000 and each element occupies only 4 bytes, find the address of the element A[2][3].

a) 1048

b) 1052

c) 1060

d) 1072

e) None of the above

26. 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) $O(n^2)$

e) More than one of the above

27. What is the time complexity for deleting a node from linked list?

a) O(1)

b) O(n)

c) O(logn)

d) More than one of the above

e) None of the above

28. 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

e) More than one of the above

29. A linear collection of data elements where the linear node is given by a pointer is called?

a) Linked list

b) Node list

c) Primitive list

d) Unordered list

e) More than one of the above

30. 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

e) More than one of the above

31. In linked list, each node contains a minimum of two fields. One field is the data field to store the data. The second field is?

a) Pointer to character

b) Pointer to integer

c) Pointer to node

d) Node

e) More than one of the above

32. 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) $\theta(n^2)$

d) More than one of the above

e) None of the above

33. 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) More than one of the above

e) None of the above

34. 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) More than one of the above

e) None of the above

35. 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) More than one of the above

e) None of the above

36. 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) More than one of the above

e) None of the above

37. Which data structure is typically used to implement hash table ?

a) Linked list

b) Array

c) Binary Tree

d) More than one of the above

e) None of the above

38. What kind of linked list is best to answer questions 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

e) More than one of the above

39. Linked lists are not suitable for the implementation of _____.

a) Insertion sort

b) Radix sort

c) Polynomial manipulation

d) Binary search

e) More than one of the above

40. Linked list is considered as an example of _____ type of memory allocation.

a) Dynamic

b) Static

c) Compile time

d) Heap

e) More than one of the above

41. In Linked List implementation, a node carries information regarding _____.

a) Data
b) Link
c) Data and Link
d) Node
e) More than one of the above

42. Linked list data structure offers considerable saving in _____.

a) Computational Time

b) Space Utilization

c) Speed Utilization

d) More than one of the above

e) None of the above

43. Which of the following points is/are not true about Linked List data structure when it is compared with an 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

e) More than one of the above

44. 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

45. 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) More than one of the above

e) None of the above

46. Given a pointer to a node X in a singly linked list. Only one pointer is given, pointer to the head node is not given. Can we delete the node X from the given linked list?

a) Possible if X is not the last node

b) Possible if the size of the linked list is even

c) Possible if the size of the linked list is odd

d) Possible if X is not the first node

e) More than one of the above

47. You are given pointers to the 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 the first element

c) Delete the last element of the list

d) Add a new element at the end of the list

e) More than one of the above

48. 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) More than one of the above

e) None of the above

49. Which of the following is false about a doubly linked list?

a) We can navigate in both 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 a singly linked list

e) More than one of the above

50. 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 a bitwise AND operator for storing addresses

e) More than one of the above

51. How do you calculate the pointer difference in a memory-efficient doubly 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

e) None of the above

52. 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) More than one of the above

e) None of the above

53. 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) In a circular linked list, each node points to the previous node instead of the next node

d) The head node is known in a circular linked list

e) More than one of the above

54. 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) $O(n^2)$

55. Which of the following applications makes use of a circular linked list?

a) Undo operation in a text editor

b) Recursive function calls

c) Allocating CPU to resources

d) Implementing hash tables

e) More than one of the above

56. 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

e) More than one of the above

57. Consider a small circular linked list. How can you effectively detect the presence of cycles in this list?

a) Keep one node as head and traverse another temp node until 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 the slow pointer advancing by one node at a time

c) Cannot determine, you have to pre-define if the list contains cycles

d) A circular linked list itself represents a cycle, so no new cycles can be generated

e) More than one of the above

58. Which of the following methods can be used to search for an element in a linked list?

a) Iterative linear search

b) Iterative binary search

c) Recursive binary search

d) Normal binary search

e) More than one of the above

59. What will be the time complexity when binary search is applied to a linked list?

a) O(1)

b) O(n)

c) $O(n^2)$

d) More than one of the above

e) None of the above

60. Which type of linked list stores the address of the header node in the next field of the last node?

a) Singly linked list

b) Circular linked list

c) Doubly linked list

d) Hashed list

e) None of the above