

Fullstack Development Milestone Test-7

1. Popular Notations in Complexity Analysis of Algorithms

- a) Big-O Notation
- b) Omega Notation
- c) Theta Notation
- d) All the above

Correct Ans: All the above

2. Which Complexity analysis is generally used?

- a) Worst Case Analysis
- b) Average Case Analysis
- c) Best Case Analysis
- d) All the above

Correct Ans: Worst Case Analysis

3. An algorithm: can be represented through _____

- a) flow charts
- b) pseudo-codes
- c) instructions in common language
- d) all of the mentioned

Correct Ans: all of the mentioned

4. There are two algorithms suppose A takes 1.41 milliseconds while B takes 0.9 milliseconds, which one of them is better considering all

other things the same?

- a) A is better than B
- b) B is better than A
- c) Both are equally good
- d) None of the mentioned

Correct Ans: B is better than A

5. For a recursive algorithm _____

- a) a base case is necessary and is solved without recursion.
- b) a base case is not necessary
- c) does not solve a base case directly
- d) none of the mentioned

Correct Ans: a base case is not necessary

6. Constant Time Complexity is

- a) $O(1)$
- b) $O(n)$
- c) $O(nc)$
- d) $O(\log \log n)$

Correct Ans: $O(1)$

7. Linear Time Complexity

- a) $O(1)$
- b) $O(n)$

c) $O(nc)$

d) $O(\text{Log Log } n)$

Correct Ans: $O(n)$

8. Quadratic Time Complexity

a) $O(1)$

b) $O(n)$

c) $O(n^c)$

d) $O(\text{Log Log } n)$

Correct Ans: $O(n^c)$

9. Logarithmic Time Complexity

a) $O(1)$

b) $O(n)$

c) $O(n^c)$

d) $O(\text{Log Log } n)$

Correct Ans: $O(\text{Log Log } n)$

10. What is Algorithm complexity and how to find it?

a)Time Factor

b)Space Factor

c)None of the above

d)All the above

Correct Ans: All the above

11. What is the observation made in the following JavaScript code?

```
var count = [1,,3];
```

- a)The omitted value takes “undefined”
- b)This results in an error
- c)This results in an exception
- d)The omitted value takes an integer value

Correct Ans: The omitted value takes “undefined”

12. The object has three object attributes namely _____

- a)Class, parameters, object’s extensible flag
- b)Prototype, class, objects’ parameters
- c)Prototype, class, object’s extensible flag
- d)Native object, Classes and Interfaces and Object’s extensible flag

Correct Ans: Prototype, class, object’s extensible flag

13. What will be the firstname and surname of the following JavaScript code?

```
var book = {  
  
  "main title": "JavaScript",  
  
  'sub-title': "The Definitive Guide",  
  
  "for": "all audiences",  
  
  author: {  
  
    firstname: "David",  
  
    surname: "Flanagan"  
  
  }  
}
```

```
};
```

- a)properties
- b)property values
- c)property names
- d)objects

Correct Ans: property names

14. To determine whether one object is the prototype of (or is part of the prototype chain of) another object, one should use the _____

- a)isPrototypeOf() method
- b>equals() method
- c)=== operator
- d)===opertor

Correct Ans: isPrototypeOf() method

15. Identify the process done in the following JavaScript code snippet?

```
o = {x:1, y:{z:[false,null,""]}};
```

```
s = JSON.stringify(o);
```

```
p = JSON.parse(s);
```

- a)Object Encapsulation
- b)Object Serialization
- c)Object Abstraction
- d)Object Encoding

Correct Ans: Object Serialization

16. What will be the output of the following JavaScript code?

```
const object1 = {};  
  
a = Symbol('a');  
  
b = Symbol.for('b');  
  
object1[a] = 'harry';  
  
object1[b] = 'derry';  
  
const objectSymbols = Object.getOwnPropertySymbols(object1);  
  
console.log(objectSymbols.l
```

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

Correct Ans: 2

17. What will be the output of the following JavaScript code?

```
const obj1 =  
  
{  
  
property1: 21  
  
}  
  
const descriptor1 = Object.getOwnPropertyDescriptor(obj1, 'property1');  
  
console.log(descriptor1.configurable);  
  
console.log(descriptor1.enumerable);
```

- a>true 21
- b>true false

c>true true

d>false false

Correct Ans: true true

18. What will be the output of the following JavaScript code?

```
const obj1 = { property1: '10'};
```

```
const obj2 = Object.freeze(obj1);
```

```
obj2.property1 = '20';
```

```
console.log(obj2.property1);
```

a)10

b)20

c)Runtime error

d)Compilation error

Correct Ans: 10

19. What will be the output of the following JavaScript code?

```
const object1 = {
```

```
property1: 20
```

```
};
```

```
console.log(Object.is(object1));
```

a)20

b)TRUE

c)FALSE

d)error

Correct Ans: FALSE

20. What will be the output of the following JavaScript code?

```
const obj = {prop: 12};
```

```
Object.preventExtensions(obj);
```

```
console.log( Object.isExtensible(obj));
```

- a)12
- b)TRUE
- c)FALSE
- d)error

Correct Ans: FALSE

21. Merge sort uses which of the following technique to implement sorting?

- a)backtracking
- b)greedy algorithm
- c)divide and conquer
- d)dynamic programming

Correct Ans: divide and conquer

22. 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) $O(n \log n^2)$

Correct Ans: $O(n \log n)$

23. What is the auxiliary space complexity of merge sort?

- a) $O(1)$
- b) $O(\log n)$
- c) $O(n)$
- d) $O(n \log n)$

Correct Ans: $O(n)$

24. Merge sort can be implemented using $O(1)$ auxiliary space.

- a) TRUE
- b) FALSE

Correct Ans: TRUE

25. 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) $O(n \log n^2)$

Correct Ans: $O(n \log n)$

26. Which of the following method is used for sorting in merge sort?

- a) merging
- b) partitioning
- c) selection

d)exchanging

Correct Ans: merging

27. 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) $O(n \log n^2)$

Correct Ans: $O(n \log n)$

28. 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

Correct Ans: linear merge sort

29. 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

Correct Ans: it is an adaptive algorithm

30. Which of the following is not in place sorting algorithm by default?

- a)merge sort
- b)quick sort
- c)heap sort
- d)insertion sort

Correct Ans: merge sort

31. Which of the following stable sorting algorithm takes the least time when applied to an almost sorted array?

- a)merge sort
- b)quick sort
- c)heap sort
- d)insertion sort

Correct Ans: merge sort

32. External-sort merge algorithm is called an ____ because it merges N runs.

- a)One-way
- b)Two-way
- c)N-way
- d)Null-way

Correct Ans: N-way

33. Choose the correct statement about bottom up merge sort from the following?

- a)bottom up merge sort has greater time complexity than standard merge sort
- b)bottom up merge sort has lesser time complexity than standard merge sort
- c)bottom up merge sort saves auxiliary space required on call stack

d)bottom up merge sort uses recursion.

Correct Ans: bottom up merge sort saves auxiliary space required on call stack

34. Choose the correct code for merge sort.

a)<pre>

```
merge_sort(arr, left, right)

{
if (left > right)

{
let mid = (right-left)/2;
merge_sort(arr, left, mid);
merge_sort(arr, mid+1, right);
merge(arr, left, mid, right); //function to merge sorted arrays
}
}
</pre>
```

b)<pre>

```
merge_sort(arr, left, right)

{
if (left < right)

{
let mid = left+(right-left)/2;
merge_sort(arr, left, mid);
merge_sort(arr, mid+1, right);
```

```
merge(arr, left, mid, right); //function to merge sorted arrays
```

```
}
```

```
}
```

```
</pre>
```

```
c)<pre>
```

```
merge_sort(arr, left, right)
```

```
{
```

```
if (left < right)
```

```
{
```

```
let mid = left+(right-left)/2;
```

```
merge(arr, left, mid, right); //function to merge sorted arrays
```

```
merge_sort(arr, left, mid);
```

```
merge_sort(arr, mid+1, right);
```

```
}
```

```
}
```

```
</pre>
```

```
d)<pre>
```

```
merge_sort(arr, left, right)
```

```
{
```

```
if (left < right)
```

```
{
```

```
let mid = (right-left)/2;
```

```
merge(arr, left, mid, right); //function to merge sorted arrays
```

```
merge_sort(arr, left, mid);
```

```
merge_sort(arr, mid+1, right);  
  
}  
  
</pre>
```

Correct Ans: <pre>

```
merge_sort(arr, left, right)  
  
{  
  if (left < right)  
  {  
    let mid = left+(right-left)/2;  
    merge_sort(arr, left, mid);  
    merge_sort(arr, mid+1, right);  
    merge(arr, left, mid, right); //function to merge sorted arrays  
  }  
}  
  
</pre>
```

35. When sorting externally, the ____ merge is the most appropriate method to use.

- a) Internal-sort
- b) External-sort
- c) Merge-sort
- d) None

Correct Ans: External-sort

36. What is the space complexity of in place merge sort?

- a) $O(1)$
- b) $O(n)$
- c) $O(\log n)$
- d) $O(n \log n)$

Correct Ans: $O(\log n)$

37. What is the average time complexity of in place merge sort when we use the following function for merging?

```
in_place_merge(arr, l, middle, r)
{
    let start2 = middle + 1;
    if (arr[middle] <= arr[start2])
    {
        return;
    }
    while (l <= middle && start2 <= r)
```

- a) $O(n \log n)$
- b) $O(n^2)$
- c) $O(n^2 \log n)$
- d) $O(n \log n^2)$

Correct Ans: $O(n^2)$

38. In place merge sort has same time complexity as standard merge sort.

a)TRUE

b)FALSE

Correct Ans: FALSE

39. In-place merge sort is a stable sort.

a)TRUE

b)FALSE

Correct Ans: TRUE

40. Choose the incorrect statement about merge sort from the following?

a)both standard merge sort and in-place merge sort are stable

b)standard merge sort has greater time complexity than in-place merge sort

c)standard merge sort has greater space complexity than in-place merge sort

d)in place merge sort has $O(\log n)$ space complexity

Correct Ans: standard merge sort has greater time complexity than in-place merge sort