

Find the longest uniform substring length and index from the given string

## **Freshworks (Chennai)**

Is the given string a valid parenthesis

Create a Basic Pub Sub Data structure

Print the names in folder structure format in the console. You can create your own static object.

## **Walmart**

Create a Polyfil for Promise functionality

Throttle Function => Write a function that should take a delay parameter and return a function.

The returned function can be executed n times. But if it has been executed 3 times, the 4th time should wait until the cooling period(delay is done).

5 calls are made in intervals 30ms, 50ms, 100ms, 130ms, 180ms to a throttle function with 30ms delay. Which calls will be executed and which ones will be skipped?

Create a Pub Sub Data structure with a timer functionality, that should take a delay as a parameter, if not present should keep 60 mins as a timer by default post which the subscriber should be removed.

/ ===== /

## **React**

React + Redux => Async operation in redux with thunk.

React Typeahead search

Todo List

React Stopwatch timer.

## **Sumologic**

React => Todo list with check box and reverse rendering (i.e newly added item to be on top)

Unique path finder - Leet [code](https://leetcode.com/problems/unique-paths/description/) - <https://leetcode.com/problems/unique-paths/description/>

## **AJIO**

React => Comments list component like Fb or insta comments, with features like Reply, Edit and Comment for each comment.

Perforce product:

1. <https://leetcode.com/problems/best-time-to-buy-and-sell-stock/description/>

Perforce Product:

1. Polyfill for promise.all

Digio:

1. [1,2,3,3,4,4,5,5,5] → find the maximum repeated element in the array (digio)
2. Execute promises parallelly
3. Execute promises sequentially
4. Write a function that takes two strings and returns True if they are one away from each other. They are one away from each other if a single operation (changing a character, deleting a character, or adding a character) will change one of the strings to the other.  
"abcde" and "abcd" are one away (deleting a character)  
"abde" and "abcde" are one away (adding a character)  
"abfde" and "abcde" are one away (replacing a character)  
"a" and "a" are one away (changing the only character 'a' to the equivalent character 'a')  
"abc" and "bcc" are NOT one away. (They are two operations away)

Saviynt

1. Currying in javascript
2. Debounce
3. create one dropdown initially with empty values add one button whenever button is clicked , make an api call, from the array of response, show the name as dropdown option in the dropdown

UI-Path:

1. <https://leetcode.com/problems/minimum-window-substring/description/>

Other Coding Apps:

Sum of elements on left equal to sum of elements on right return that element as well as index

Sales Force:

Hacker Rank:

1. Number of projects - 3 to 4  
Project ids - [2,1,0,2]  
Bids - [9, 6, 7, 8]
  - a.  $8 + 6 + 7 = 21$
  - b. Return -1
2. Leet Code [Link](https://leetcode.com/problems/subarray-product-less-than-k/description/) - <https://leetcode.com/problems/subarray-product-less-than-k/description/>

function countSubArrayProductLessThanOrEqualK(a, k) {

let n = a.length;

```

let p = 1;
let res = 0;
for (let start = 0, end = 0; end < n; end++) {
    // Move right bound by 1 step. Update the product.
    p *= a[end];

    // Shrink the window from the left as long as the product is greater than k.
    while (start <= end && p > k) {
        p /= a[start];
        start++;
    }

    // If p is less than or equal to k, update the counter.
    // Every subarray ending at 'end' and starting from any index in [start, end] is valid.
    if (p <= k) {
        res += (end - start + 1);
    }
}

return res;
}

// console.log('res:', countSubArrayProductLessThanOrEqualK([2, 3, 4], 6));
// console.log('res:', countSubArrayProductLessThanOrEqualK([3,1,2,3,7], 6));
console.log('res:', countSubArrayProductLessThanOrEqualK([1,2,3], 7));

```

1. LRU Cache (coding question)
2. Chess game design (design round HLD)

## Theory

1. How do you achieve Responsive web design
2. How do you achieve Accessibility in a webpage
3. Explain uses of Sass , mixin
4. Repaint and reflow
5. Pre-fetch
6. Preload
7. How to Optimize performance of the web page
8. Observer design pattern
9. Factory design pattern
10. Iterators, generators
11. Yield

12. Abstract classes (no new object can be created from the abstract class, it is only used for inheritance)
13. Weak map
14. Different types in typescript
15. Generic classes
16. Omit
17. Union
18. Generator functions
19. Which mechanism is responsible for the change detection in angular (zone.js in the older version)
20. Multicasting in angular
21. Merge map, switch map and concat map in
22. Latest features in angular
23. Si

#### Angular Interview Questions

1. Types of forms i.e Reactive forms and template driven forms
2. How to create dynamic forms
3. Pipes, pure pipes and impure pipes
4. Directives, custom directives
5. How to improve the performance of angular application
6. Encapsulation in angular
7. Latest features in angular
8. Lazy loading in angular
9. Life cycle hooks
10. Content projection in angular
11. Change detection strategies in angular
12. View children and view child
13. Content child and content children
- 14.