

# Twitter Cracker Sheet

- By Sunyul Hossen

Note: [To Use this sheet optimally go and watch the "Cracking the Twitter coding interview🔥: The definitive prep guide" video on Debug Buzz Channel.

Link - <https://youtu.be/-xM32sTWdhc> ]

**Q)**

```
import java.util.*;

public class Solution {

    public static final class MapDiff {

        /**
         * Given two maps, return the "difference" between them
         * What's a good definition of difference?
         * What's a good return type for this method?
         */

        public static <K, V> ??? diff(Map<K, V> left, Map<K, V> right) {

        }

    }

    public static void main(String[] args) {

    }

}
```

**Q)**

Given a list of sentences build a suggestion engine, where if we provide an input word it returns the most frequent next word from the input list.

For Example:

[

["Alice", "jumped", "into", "the", "hole"],

["A", "rabbit", "jumped", "into", "the", "hole"],

["The", "king", "sent", "forth", "his", "troops"],

.....

]

In this particular example if "the" is the input word, the suggestion should be "hole" because, ["the", "hole"] appeared twice in the list of sentences where as ["the", "king"] appeared only once.

If "hole" is the input we return empty string because no suggestion exists.

**Q)**

Suppose two players A and B play ping pong. The first player to get to 11 points wins the match. The players alternate after every 2nd serve (e.g. player A serves twice first, then player B serves twice and so on).

If A is serving, the probability of A winning the point is  $p_A$ .

If B is serving, the probability of B winning the point is  $p_B$ .

If player A serves first, what is the probability of player A winning the match? Can you determine the solution algorithmically?

**Q)**

You are given a list of "bad expressions" ahead of time.

You can preprocess them, or do whatever you need.

You are to implement a method that receives a tweet, and tries to find any of the bad expressions in the tweet, and to return a list of all the bad expression you found.

Example:

Bad expressions:

foo

food

frank

this has

is has

return

now now

Example tweet:

"this has been #TK421Tweets. We now return you to your regularly scheduled Twitter."

Result:

this has

return

is has

**Q)**

// Write a method to find out how much total time was spent in each space,

// over all users.

// action, space, user, timestamp

static String [][] requests = {

    {"create", "xyz", "1", "1619916081"}, // Sat May 01 2021 17:41:21 GMT-0700

    {"join", "xyz", "2", "1619916681"}, // Sat May 01 2021 17:51:21 GMT-0700

    {"create", "abc", "3", "1619916881"}, //12:01

    {"leave", "xyz", "2", "1619920281"},

    {"join", "abc", "4", "1619920881"},

    {"create", "ghi", "5", "1619923999"},

    {"leave", "xyz", "1", "1619923881"},

    {"leave", "abc", "3", "1619927481"}, //12:07 time spent = 6 minutes

    {"leave", "abc", "4", "1619927481"},

    {"leave", "ghi", "5", "1619958001"}

};

For Userid = 3, total time spent is 12:07 -12:01=6 mins

**Q)**

given a list of login requests [timestamp, IP] sorted by timestamp in ascending order, detect all IPs with more than k login requests in the window time

**Q)**

Find all unique combinations of an input string of size m.

inputStr = "ABCDE"

size = N

When size=2, output is =

[AA,AB, AC, AD, AE, BB, BA, ....]

**Q)**

HackerCards is a trendy new card game. Each type of HackerCard has a distinct ID number greater than or equals to 1, and the cost of each HackerCard equals its ID number. For example, HackerCard 1 costs 1, HackerCard 5 costs 5, and so on.

Leanne already has a collection started. For her birthday, Mike wants to buy her as many cards as he can, but they must cost less than or equals to his budget. He wants to buy one each of some cards she doesn't already have. If he has to make one choice among several, he will always choose the lowest cost option. Determine which cards he will buy.

For example, Leanne's collection = [2, 4, 5] and Mike has d = 7 to spend. He can purchase a maximum of 2 cards, the 1 and the 3 to add to her collection. Two other options he has are 1 and 6 (costs more) or 7(fewer cards, costs more)

Input:

collections[] = [1,3,4]

d = 7

output: [2,5]

**Q)**

Given an array of words and an array of sentences, determine which words are anagrams of each other. Calculate how many sentences can be created by replacing any word with one of its anagrams

Input:

```
wordSet = ['the', 'bats', 'tabs', 'in', 'cat', 'act']  
sentences = ['cat the bats', 'in the act', 'act tabs in']
```

output: [4,2,4]

**Q)**

Given a string, reduce it in such a way that all of its substrings are distinct. To do so, you may delete any characters at any index. What is the minimum number of deletions needed?

**Q)**

Given a string that consists of left and right parentheses, '(' and ')', balance the parentheses by inserting parentheses as necessary. Determine the minimum number of characters that must be inserted.

Input:

```
S = ' ( ) ) ) '
```

output: 2

**Q)**

# 1. Checking Your Route

A ride hailing company sometimes travels between cities. To avoid delays, a driver first checks for the shortest routes. Given a map of the cities and their bidirectional roads represented by a graph of nodes and edges.

**Determine the paths from the first node to the last node and choose the shortest length.**

**Now select all paths that are that length.**

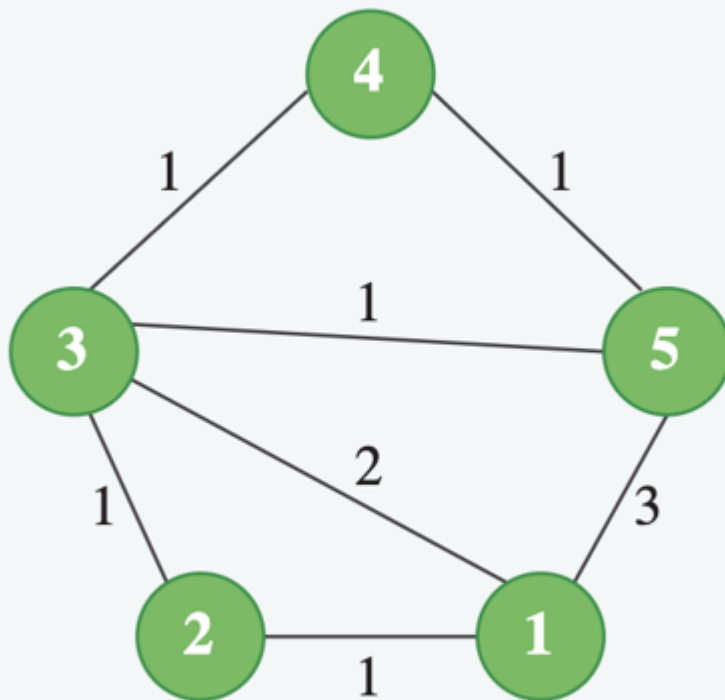
**These are the shortest paths.** Return an array of strings, one for each road in order, where the value is YES if the road is along a shortest path or NO if it is not. The roads or *edges* are named using their *1-based* index within the input arrays.

## Example

given a map of  $g\_nodes = 5$  nodes, the starting nodes, ending nodes and road lengths are:

Road	
from/to/weight	
1	(1, 2, 1)
2	(2, 3, 1)
3	(3, 4, 1)
4	(4, 5, 1)
5	(5, 1, 3)
6	(1, 3, 2)
7	(5, 3, 1)

It is represented in the graph below:



- The traveller must travel from city 1 to city  $g\_nodes$ , so from node 1 to node 5 in this case.
- The shortest path is 3 units long and there are three paths of that length:  $1 \rightarrow 5$ ,  $1 \rightarrow 2 \rightarrow 3 \rightarrow 5$ , and  $1 \rightarrow 3 \rightarrow 5$ .
- Return an array of strings, one for each road in order, where the value is YES if a road is along a shortest path or NO if it is not. In this case, the resulting array is ['YES', 'YES', 'NO', 'NO', 'YES', 'YES', 'YES']. The third and fourth roads connect nodes (3, 4) and (4, 5) respectively. They are not on a shortest path, i.e. one with a length of 3 in this case.

### Function Description

Complete the function `classifyEdges` in the editor below.

classifyEdges has the following parameter(s):

*int g\_nodes*: an integer, the number of nodes

*int g\_from[g\_edges]*: an array of integers, the start g\_nodes for each road

*int g\_to[g\_edges]*: an array of integers, the end g\_nodes for each road

*int g\_weight[g\_edges]*: an array of integers, the lengths of each road

### Returns

*string[g\_edges]*: an array of strings where the value at  $i^{\text{th}}$  index is YES if the  $i^{\text{th}}$  edge is a part of a shortest path from vertex 1 to vertex g\_nodes. Otherwise it should contain NO.

### Constraints

- $2 \leq g\_nodes \leq 3000$
- $1 \leq g\_edges \leq \min(10^5, (g\_nodes \times g\_nodes - 1)/2)$
- $1 \leq g\_weight[i] \leq 10^5$
- $1 \leq g\_from[i], g\_to[i] \leq g\_nodes$
- There is at most one edge between any pair of g\_nodes
- The given graph is connected



## Sample Input 0

```
STDIN          Function Parameters
-----          -
4 5  →        g_nodes = 4   g_edges = 5
1 2 1 →       g_from = [1, 2, 1, 3, 1]
g_to = [2, 4, 3, 4, 4] g_weight=[1,
1, 1, 2, 2]
2 4 1
1 3 1
3 4 2
1 4 2
```

## Sample Output 0

```
YES
YES
NO
NO
YES
```

Q)

1. Partitioning array:  
[https://nataliekung.gitbook.io/ladder\\_code/twitter-oa/partitioning-array](https://nataliekung.gitbook.io/ladder_code/twitter-oa/partitioning-array)
2. Paint house: [https://nataliekung.gitbook.io/ladder\\_code/twitter-oa/paint-house](https://nataliekung.gitbook.io/ladder_code/twitter-oa/paint-house)
3. Palindrom substring:  
[https://nataliekung.gitbook.io/ladder\\_code/twitter-oa/palindromic-substrings](https://nataliekung.gitbook.io/ladder_code/twitter-oa/palindromic-substrings)
4. Twitter user network: given a matrix  $n * n$ , where the  $(i, j)$  entry is either 1 or 0, and a 1 represents that user  $i$  follows user  $j$ . If  $i$  follows  $j$  or  $j$  follows  $i$ , they are in the same group. Return the total number of groups.

Q)

### 3. String Reduction

---

Given a string, reduce it in such a way that all of its substrings are distinct. To do so, you may delete any characters at any index. What is the minimum number of deletions needed?

**Note:** A *substring* is a contiguous group of 1 or more characters within a string.

#### Example

$s = \text{"abab"}$

Substrings in  $s$  are { 'a', 'b', 'a', 'b', 'ab', 'ba', 'ab', 'aba', 'bab', 'abab'}. By deleting one "a" and one "b", the string becomes "ab" or "ba" and all of its substrings are distinct. This required 2 deletions.

#### Function Description

Complete the function *getMinDeletions* in the editor below.

*getMinDeletions* has the following parameter(s):

string  $s$ : the given string

Returns:

int: the minimum number of deletions required

#### Constraints

- $1 \leq n \leq 10^5$

# 1. Even Subarray

A subarray is a contiguous portion of an array. Given an array of integers, determine the number of distinct subarrays that can be formed having at most a given number of odd elements. Two subarrays are distinct if they differ at even one position in their contents.

## Example

*numbers* = [1, 2, 3, 4]

*k* = 1

The following is a list of the 8 distinct valid subarrays having no more than 1 odd element:

```
[[1], [2], [3], [4], [1,2], [2, 3], [3, 4], [2, 3, 4]]
```

## Function Description

Complete the function *evenSubarray* in the editor below.

*evenSubarray* has the following parameter(s):

*int numbers[n]*: an array of integers

*int k*: the maximum number of odd elements that can be in a subarray

## Return

*int*: the number of distinct subarrays that can be formed as described

## Constraints

- $1 \leq n \leq 1000$
- $1 \leq k \leq n$
- $1 \leq \text{numbers}[i] \leq 250$

## 4. The Largest String

Given a string, construct a new string by rearranging the original string and deleting characters as needed. Return the alphabetically largest string that can be constructed respecting a limit as to how many consecutive characters can be the same.

### Example

$s = \text{'baccc'}$

$k = 2$

The largest string, alphabetically, is  $\text{'cccba'}$  but it is not allowed because it uses the character  $\text{'c'}$  more than 2 times consecutively. Therefore, the answer is  $\text{'ccbca'}$ .

### Function Description

Complete the function *getLargestString* in the editor below.

*getLargestString* has the following parameters:

string  $s[n]$ : the original string

int  $k$ : the maximum number of identical consecutive characters the new string can have

Returns:

*string*: the alphabetically largest string that can be constructed that has no more than  $k$  identical consecutive characters

### Constraints

- $1 \leq n \leq 10^5$
- $1 \leq k \leq 10^3$
- The string  $s$  contains only lowercase English letters.

## 2. How many Sentences ?

Given an array of words and an array of sentences, determine which words are anagrams of each other. Calculate how many sentences can be created by replacing any word with one of its anagrams.

### Example

*wordSet* = ['listen', 'silent', 'it', 'is']

*sentence* = 'listen it is silent'

Determine that *listen* is an anagram of *silent*. Those two words can be replaced with their anagrams. The four sentences that can be created are:

- listen it is silent
- listen it is listen
- silent it is silent
- silent it is listen

### Function Description

Complete the *countSentences* function in the editor below.

*countSentences* has the following parameters:

*string wordSet[n]*: an array of strings

*string sentences[m]*: an array of strings

Returns:

*int[]*: an array of *m* integers that denote the number of sentences that can be formed from each sentence

### Constraints

- $0 < n \leq 10^5$
- $1 \leq \text{length of each word} \leq 20$
- $1 \leq m \leq 1000$
- $3 \leq \text{words in a sentence} \leq 20$

**Q)**

Given an array of integers, determine the number of moves to make all elements equal.

Each move consists of choosing all but 1 element and incrementing their values by 1

Eg: [3,4,6,6,3]

Output: 7

**Q)**

Disk Space Analysis : Given nums and segment x. Find for each segment of size x the minimum number and find the maximum of all these segments.

Eg : [ 8,2,4,6 ] , x = 2

output = max( [ 2 , 2 , 4 ] ) = 4

**Q)**

Conference schedule:-

<https://leetcode.com/problems/maximum-number-of-events-that-can-be-attended/>

**Q)**

Given an array of words and array of sentences, determine which words are anagrams of each other. Calculate how

many sentences can be created by replacing any word with one of its anagram.

wordset = ["listen", "silent", "it", "is"]

sentence = "listen it is silent"

**Q)**

- Coloring the Blocks
- Balanced Sales array
- Reaching Points
- University Career Fair

**Q)**



5m left

### 1. Weird Faculty

This semester you are taking a course taught by a faculty member who has a weird way of grading tests. In a test,  $n$  questions will be asked, and the correctness of the answers is already determined. For the  $i^{\text{th}}$  question, the verdict will be  $v[i]$  (where  $0 \leq i < n$ ). If  $v[i] = 1$ , the answer is correct but if  $v[i] = 0$ , the answer is wrong.

When a test is given, you have to answer the first  $k$  questions (indices  $0$  to  $k-1$ ) where  $0 \leq k < n$ , and your friend has to answer the remaining questions (indices  $k$  to  $n-1$ ) on the test. At the end, results are calculated as follows:

<pre>int Your_result = 0; for(int i=0;i&lt;k;i++) {     if(v[i]==1)         Your_result++;     else Your_result--; }</pre>	<pre>Your friend's results: int YourFriend d_result = 0; for(int i=k;i&lt;n;i ++) {     if(v[i]== 1)</pre>
--	--

```

10
11 /*
12  * Complete the 'exam' function below.
13  * The function is expected to return an INTEGER.
14  * The function accepts INTEGER_ARRAY v as parameter.
15  */
16
17 int exam(vector<int> v) {
18     int n=v.size();
19     int k,p=0;
20
21     for(k=0;k<n;k++)
22     {
23
24         int Your_result;
25         for(int i=0;i<k;i++)
26         {
27             if(v[i]==1)
28                 Your_result++;
29             else Your_result--;
30         }
31         int my_frnd;
32         for(int i=k;i<n;i++)
33         {
34             if(v[i]==1)
35                 my_frnd++;
36             else my_frnd--;
37         }
38         if(Your_result>my_frnd)
39         {
40             ++(n)=k;

```

Line: 9 Col: 1

Test Results Custom Input Run Code Run Tests Submit

10m left

1) YourFriend d\_result+; else YourFriend d\_result-;

Choose the minimum  $k$  such that  $Your\_result > YourFriend\_result$ .

#### Function Description

Complete the function `exam` in the editor below. The function must return an integer that denotes the minimum number of questions you must answer to have  $Your\_result > YourFriend\_result$ .

`exam` has the following parameter(s):

- `v[0]...v[n-1]`: an array of integers

#### Constraints

- $1 \leq n \leq 10^5$
- $v[i] \in \{0,1\}$  (where  $0 \leq i < n$ )

Input Format For Custom Testing

Sample Case 0

Sample Input 0

```

77 int result = exam(v);
78
79 cout << result << "\n";
80
81 fcout.close();
82
83 return 0;
84 }
85
86 string ltrim(const string &str) {
87     string s(str);
88
89     s.erase(
90         s.begin(),
91         find_if(s.begin(), s.end(), not1(ptr_fun<int, int>(isspace)))
92     );
93     return s;
94 }
95
96 string rtrim(const string &str) {
97     string s(str);
98
99     s.erase(
100         find_if(s.rbegin(), s.rend(), not1(ptr_fun<int, int>(isspace))).base(),
101         s.end()
102     );
103     return s;
104 }
105
106 }
107

```

Line: 76 Col: 1

Test Results Custom Input Run Code Run Tests Submit

Q)



2. K-Difference

Given an array of distinct integers and a target value, determine the number of distinct pairs of integers in the array with an absolute difference equal to the target amount. Two pairs are distinct if they differ in at least one value. To illustrate, the pairs (1, 3) and (3, 2) are distinct. The pairs (1, 3) and (3, 1) are not.

For example, given the array  $a = [1, 3, 5]$  and a target difference  $k = 2$ . There are two pairs:  $[1, 3]$  and  $[3, 5]$ , that have the target difference.

**Function Description**  
Complete the function `kDifference` in the editor below. The function must return an integer that represents the number of distinct pairs in `a` having a difference of `k`.

`kDifference` has the following parameter(s):  
`a[a[0]...a[n-1]]`: array of integers  
`k`: the target difference

**Constraints**

- $5 \leq n \leq 10^5$
- $0 < a[i] \leq 2 \times 10^9$
- Each `a[i]` is distinct, i.e. unique within `a`.
- $1 \leq k \leq 10^9$

```

1 > #include <bits/stdc++.h>...
9
10 /*
11  * Complete the 'kDifference' function below.
12  *
13  * The function is expected to return an INTEGER.
14  * The function accepts following parameters:
15  * 1. INTEGER_ARRAY a
16  * 2. INTEGER k
17  */
18 int kDifference(vector<int> a, int k) {
19     int end;
20     for(int begn=0;begn<a.size();begn++)
21         int end=a.size()-1;
22         { while(end!=begn)
23             a[begn]
24         }
25
26 }
27
28 > int main()...

```

Test Results Custom Input Run Code Run Tests Submit

Q)

3. Get set go

Kane is a Twitter employee and a fitness freak. But, it is tough to say no to the culinary delights of the Twitter chefs. Twitter chefs also mention the calorie value of a meal they are going to prepare for the coming days in advance. So, Kane came up with a plan to be on fast some days and eat on the other days. When he is on fast, he consumes no calories, but on the day he eats, he consumes the amount of calories mentioned in the chart by the chef.

His fitness goal is the total amount of calories intake over a certain days of time, that is, he wants to consume a total of `X` number of calories EXACTLY. Let's say there is an array of positive integers representing calorie counts for each day, help Kane to decide if it's possible to eat the exact number of calories according to his fitness goal

Example - 1: Input = `calCounts : { 2, 9, 5, 1, 6}`,  
requiredCals : 12 Calories  
Output = True  
Because, he can eat on days - 0,1,3 (2 + 9 + 1 = 12) OR he can eat on days - 2,3,4 (5 + 1 + 6)

Example - 2: Input = `calCounts : { 2, 3, 15, 1, 16}`,  
requiredCals : 8 Calories  
Output = False  
Because, the target of 8 cannot be met by adding any of the subset of the values.

Limits : Size of the Array  $\leq 30$

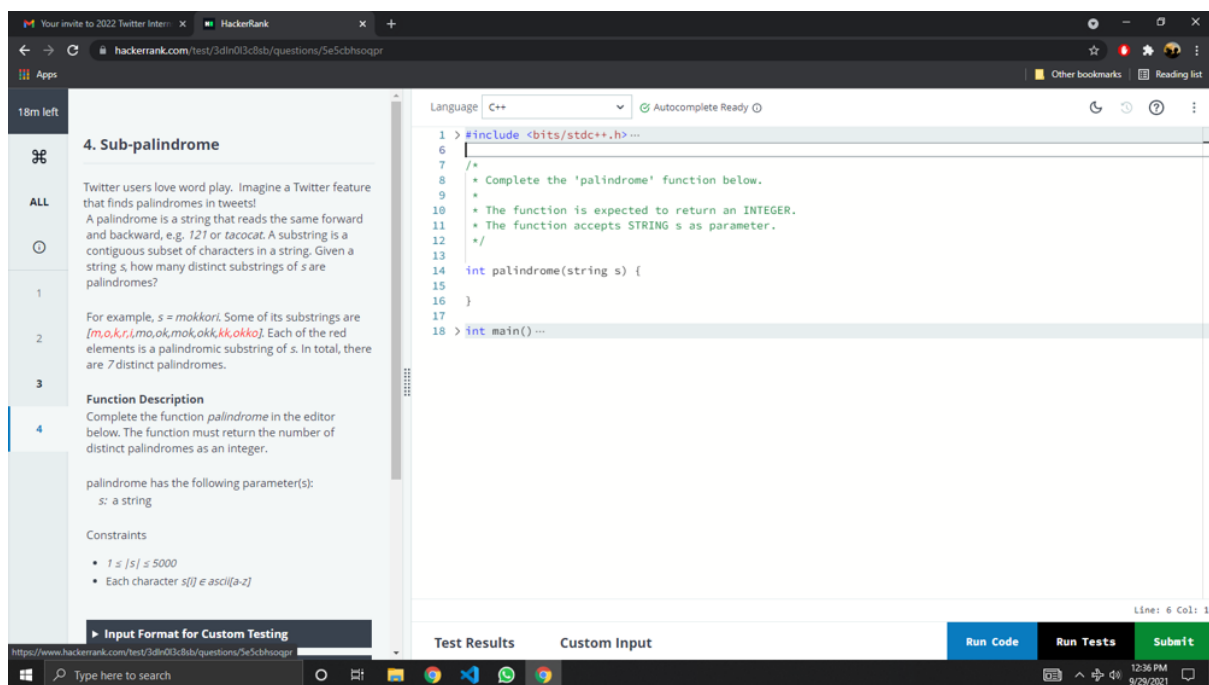
```

1 > #include <bits/stdc++.h>...
9
10 /*
11  * Complete the 'isPossible' function below.
12  *
13  * The function is expected to return a BOOLEAN.
14  * The function accepts following parameters:
15  * 1. INTEGER_ARRAY calCounts
16  * 2. INTEGER requiredCals
17  */
18
19 bool isPossible(vector<int> calCounts, int requiredCals) {
20
21 }
22
23 > int main()...

```

Test Results Custom Input Run Code Run Tests Submit

Q)



Q)

Some of the 2021 oa questions resources from leetcode:

<https://leetcode.com/discuss/interview-question/1491637/Twitter-or-Coding-Round-or-Closest-Numbers>

<https://leetcode.com/discuss/interview-question/1488062/Twitter-2022-OA-Early-Career>

<https://leetcode.com/discuss/interview-question/1491641/Twitter-or-Coding-Round-or-Array-Game>

<https://leetcode.com/discuss/interview-question/1496982/twitter-oa-early-career-2022>

<https://leetcode.com/discuss/interview-question/1495706/twitter-oa-2022-twitter-early-career-engineering-coding-challenge>

<https://leetcode.com/discuss/interview-question/1499675/twitter-oa-early-career-2022>

<https://leetcode.com/discuss/interview-question/1033712/Twitter-OA-2021-New-Grad/830572>

<https://algonmonster.medium.com/twitter-online-assessment-questions-2021-twitter-oa-hackerrank-3a62863814d4>

Questions before oa conducted in hackerrank:

<https://leetcode.com/discuss/interview-question/425440/Twitter-OA-2019-Questions>

This is the resource that helped me a lot and this is a very good resource also see discussions there you can find good resources.

[https://leetcode.com/discuss/interview-question/1346837/Decryption\\_using\\_key\\_phrases-of-HackerRank-twitter](https://leetcode.com/discuss/interview-question/1346837/Decryption_using_key_phrases-of-HackerRank-twitter)

<https://leetcode.com/discuss/interview-question/1343299/Missed-Chapters-HackerRank-twitter-and-HCI>

<https://leetcode.com/discuss/interview-question/1467769/Twitter-Stickers-problem>

<https://leetcode.com/discuss/interview-question/1487965/Twitter-Coding-Question-Vacate-hosts-in-Rack>

<https://www.1point3acres.com/bbs/collection/232531>

## Q)

Some of the frequent twitter questions(Last 6 months):

**Tweet Counts Per Frequency (38 times):**

<https://leetcode.com/problems/tweet-counts-per-frequency/>

**Design Log Storage System (8 times):**

<https://leetcode.com/problems/design-log-storage-system/>

**Insert Interval (7 times):** <https://leetcode.com/problems/insert-interval/>

**Maximum Number of Occurrences of a Substring (7 times):**

<https://leetcode.com/problems/maximum-number-of-occurrences-of-a-substring/>

**Reaching Points (7 times):** <https://leetcode.com/problems/reaching-points/>

**Insert Delete GetRandom O(1) (7 times):**

<https://leetcode.com/problems/insert-delete-getrandom-o1/>

**Maximal Square (6 times):** <https://leetcode.com/problems/maximal-square/>

**K-diff Pairs in an Array (5 times):** <https://leetcode.com/problems/k-diff-pairs-in-an-array/>

**The kth Factor of n (4 times):** <https://leetcode.com/problems/the-kth-factor-of-n/>

**Two Sum (4 times):** <https://leetcode.com/problems/two-sum/>

**Minimum Increment to Make Array Unique (4 times):**

<https://leetcode.com/problems/minimum-increment-to-make-array-unique/>

**Find Median from Data Stream (4 times):**

<https://leetcode.com/problems/find-median-from-data-stream/>

**Find Pivot Index (3 times):** <https://leetcode.com/problems/find-pivot-index/>

**Minimum Swaps to Group All 1's Together (3 times):**

<https://leetcode.com/problems/minimum-swaps-to-group-all-1s-together/>

**Meeting Rooms II (3 times):** <https://leetcode.com/problems/meeting-rooms-ii/>

**The Skyline Problem (3 times):** <https://leetcode.com/problems/the-skyline-problem/>

**Word Search II (3 times):** <https://leetcode.com/problems/word-search-ii/>

**Merge Intervals (3 times):** <https://leetcode.com/problems/merge-intervals/>

**Merge k Sorted Lists (3 times):** <https://leetcode.com/problems/merge-k-sorted-lists/>

Q)

1. <https://leetcode.com/discuss/interview-question/989768/amazon-oa-2020-transactions-logs>
2. <https://leetcode.com/problems/maximal-square/>
3. <https://leetcode.com/problems/minimum-add-to-make-parentheses-valid/>
4. String Reduction : Given a string , reduce string such that all of its substrings are distinct.  
Eg : s = "abab" substr = { a , b , ab , ba , aba , bab , abab }.  
output : Delete one 'a' and one 'b' - > return 2

## A Compilation of the Most Commonly Asked Twitter Interview Questions

Here is a compilation of all the commonly asked technical interview questions revolving around the different aspects of the particular job role.

### Twitter System Design Interview Questions

- Which coding language would you say you're adept in?
- Define SAN and explain its uses.
- When is the right time to denormalize a database design?
- Explain the impact of continuous integration systems in the automated-build process.
- Explain how you would improve the news feed on Twitter.

### Twitter Bootstrap Interview Questions

- Explain Bootstrap and its advantages.
- Discuss the key components of Bootstrap.
- Explain Bootstrap Container and how it works.
- What do you know about the default Bootstrap text settings?
- Explain the Bootstrap Grid System.
- What are the differences between Bootstrap 4 and Bootstrap 5?

- Explain the differences between Bootstrap 3 and Bootstrap 4.
- Define Button Group and the class for a basic Button Group.
- How would you use Bootstrap to make thumbnails?
- What are the two codes for displaying code in Bootstrap?

## Twitter Coding Interview Questions

- Write any YAML parser.
- Print the Fibonacci series without using an array.
- How to find the lowest common ancestor of a Binary Tree?
- How to clone a graph and analyze the time and space complexity.
- Create a unique hash function for every single tweet on Twitter.
- Find whether a directed graph contains cycles or not. Create a function with Boolean return type for the same.
- Display pattern matching using patterns containing chars.
- What is the most optimized way to sort a k-sorted array?
- What do you know about C++'s problem with multiple inheritances?
- Solve N-Queens problem.

## Twitter Behavioral Interview Questions

- Do you prefer to work alone, or are you a team player?
- What if you were asked to do a task but are not sure how to complete it?
- How to overcome the challenge of working with a difficult colleague on a team project?
- Tell me about your biggest strength.
- Describe your leadership skills.