

I participated in three different contests ranging from easy (div3) all the way to division1. The most common occurring issue I had was converting an initial solution into something that met the run time requirement. An area I aim to improve is studying/understanding tools for improving efficiency. What I mean by this is a better understanding of certain data structures. I had a very hard time converting a solution into a data structure that utilizes the problem structure efficiently.

Problem #1. Hash Table Related

Given a string containing digits from 2-9 inclusive, return all possible letter combinations that the number could represent. Return the answer in **any order**.

A mapping of digit to letters (just like on the telephone buttons) is given below. Note that 1 does not map to any letters.

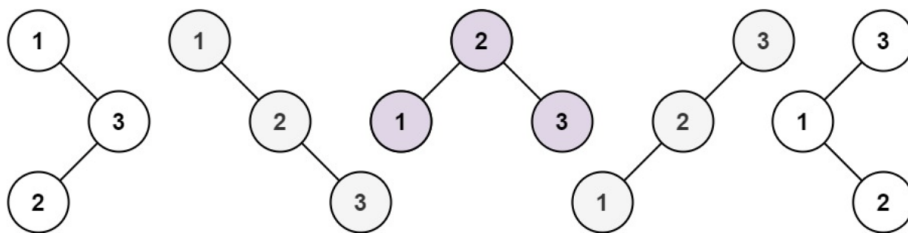


<https://leetcode.com/problems/letter-combinations-of-a-phone-number/>

Problem #2 Binary Tree Related

Given an integer n , return all the structurally unique **BST**'s (binary search trees), which has exactly n nodes of unique values from 1 to n . Return the answer in **any order**.

Example 1:



Input: $n = 3$

Output: `[[1,null,2,null,3],[1,null,3,2],[2,1,3],[3,1,null,null,2],[3,2,null,1]]`

<https://leetcode.com/problems/unique-binary-search-trees-ii/>

Problem #3 Graph Related

133. Clone Graph

Medium 4100 2001 Add to List Share

Given a reference of a node in a **connected** undirected graph.

Return a **deep copy** (clone) of the graph.

Each node in the graph contains a value (`int`) and a list (`List[Node]`) of its neighbors.

```
class Node {
    public int val;
    public List<Node> neighbors;
}
```

Test case format:

For simplicity, each node's value is the same as the node's index (1-indexed). For example, the first node with `val == 1`, the second node with `val == 2`, and so on. The graph is represented in the test case using an adjacency list.

An **adjacency list** is a collection of unordered **lists** used to represent a finite graph. Each list describes the set of neighbors of a node in the graph.

The given node will always be the first node with `val = 1`. You must return the **copy of the given node** as a reference to the cloned graph.

<https://leetcode.com/problems/clone-graph/>

Competitions completed:

Division 1 (Got destroyed)

The screenshot shows the Codeforces website interface during the 'Codeforces Round #751 (Div. 1)' contest. The top navigation bar includes links for HOME, TOP, CONTESTS, GYM, PROBLEMSET, GROUPS, and RATING. The user's profile 'gavin.taylor.mcroy' is visible in the top right corner. A notification bubble in the center states: 'The coding phase of "Codeforces Round #751 (Div. 1)" is finished, reload the page to view changes'. The main content area displays a list of problems (A-F) with their names, types, and constraints. The contest status is 'Contest is running' with a timer at 00:00:00. A 'Clone Contest to Mashup' button is visible in the bottom right corner.

#	Name	Type	Time	Memory	Flags
A	Array Elimination	standard input/output	2 s	512 MB	🚩 ⭐
B	Frog Traveler	standard input/output	2 s	512 MB	🚩 ⭐
C	Optimal Insertion	standard input/output	3 s	512 MB	🚩 ⭐
D	Difficult Mountain	standard input/output	2 s	512 MB	🚩 ⭐
E	Phys Ed Online	standard input/output	2 s	512 MB	🚩 ⭐
F	Two Sorts	standard input/output	4 s	512 MB	🚩 ⭐

[Complete problemset](#)

Codeforces Round #751 (Div. 1)

Contest is running

00:00:00

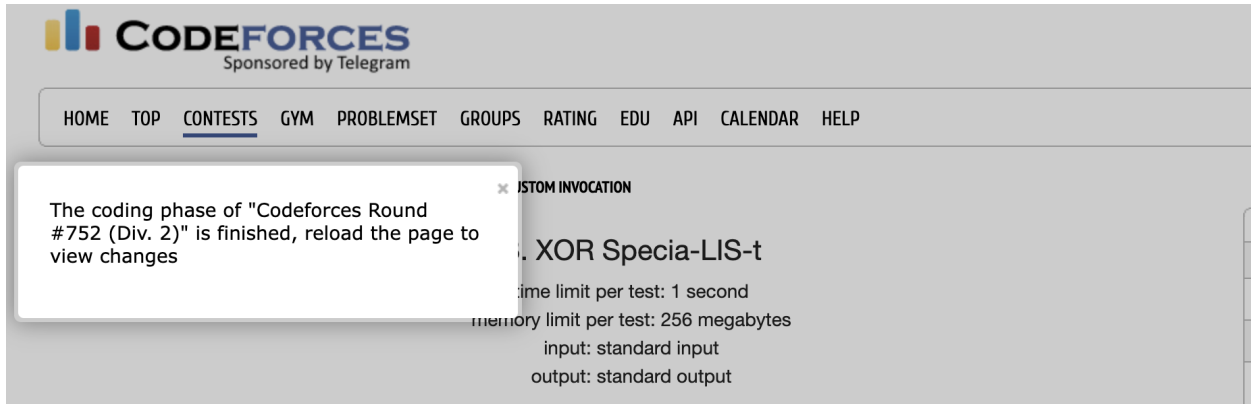
Virtual Participation

→ **Clone Contest to Mashup**

You can clone this contest to a mashup.

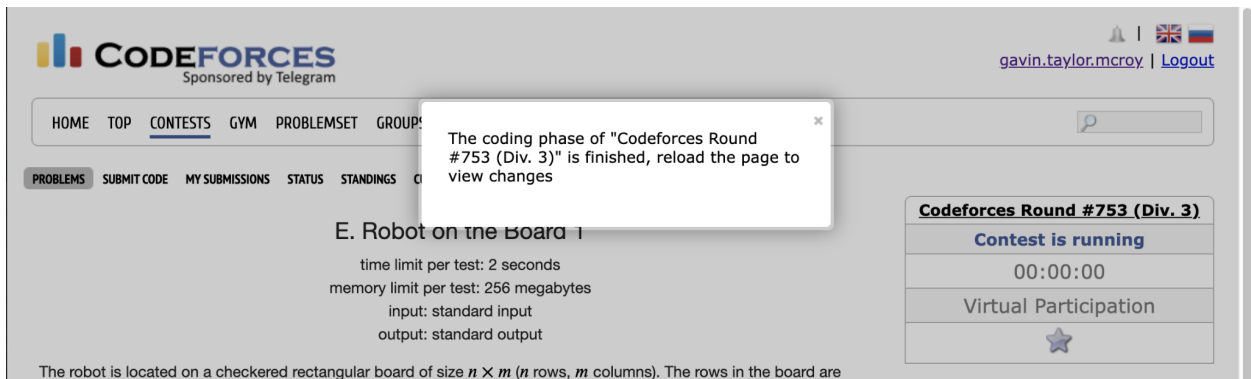
[Clone Contest](#)

Division 2 (Could come up with brute force algos but not fast ones)



The screenshot shows the Codeforces website during a contest. The header includes the Codeforces logo, "Sponsored by Telegram", and navigation links: HOME, TOP, CONTESTS, GYM, PROBLEMSET, GROUPS, RATING, EDU, API, CALENDAR, HELP. A notification box in the top left states: "The coding phase of 'Codeforces Round #752 (Div. 2)' is finished, reload the page to view changes". The main content area displays the problem title "XOR Special LIS-t", a time limit of 1 second, a memory limit of 256 megabytes, and standard input/output. A small "CUSTOM INVOCATION" button is visible near the notification.

Division 3 (Not bad)



The screenshot shows the Codeforces website during a contest. The header includes the Codeforces logo, "Sponsored by Telegram", and navigation links: HOME, TOP, CONTESTS, GYM, PROBLEMSET, GROUPS. A user profile for "gavin.taylor.mcroy" is visible in the top right, along with flags for the United Kingdom and Russia, and a "Logout" link. A notification box in the top left states: "The coding phase of 'Codeforces Round #753 (Div. 3)' is finished, reload the page to view changes". The main content area displays the problem title "E. Robot on the Board", a time limit of 2 seconds, a memory limit of 256 megabytes, and standard input/output. The bottom of the problem description reads: "The robot is located on a checkered rectangular board of size $n \times m$ (n rows, m columns). The rows in the board are". On the right side, a sidebar for "Codeforces Round #753 (Div. 3)" shows "Contest is running", a timer at "00:00:00", "Virtual Participation", and a star icon.