

DSA Assignment - 11 July

Name: Aditya Tiwari

Roll No: 23/11/EC/040

LeetCode ID: adityatiwari1305t

GeeksForGeeks ID: aditya1305t

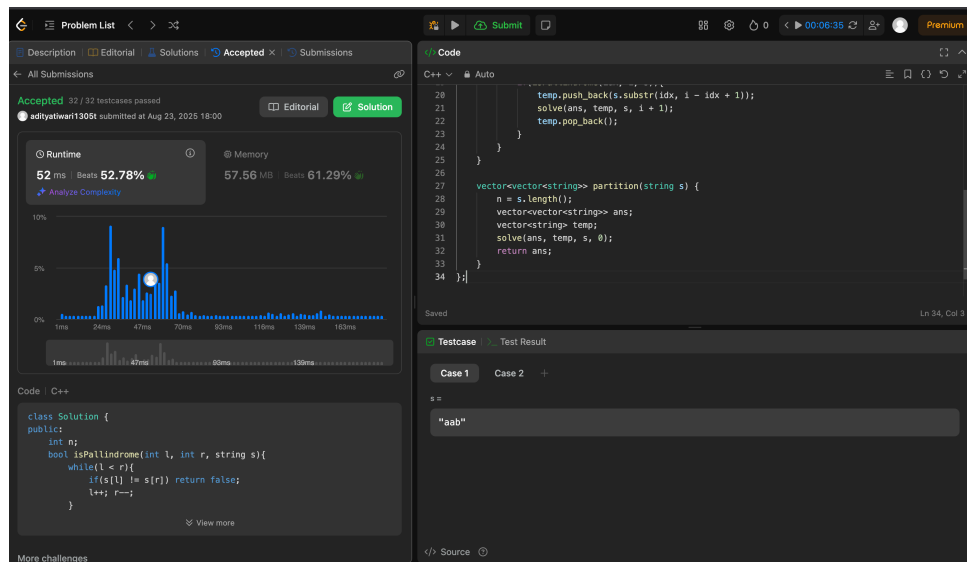
Github Repo Link:

https://github.com/Aditya1305T/SOE_Training_25

Question 1: Palindrome Partitioning

Platform: LeetCode

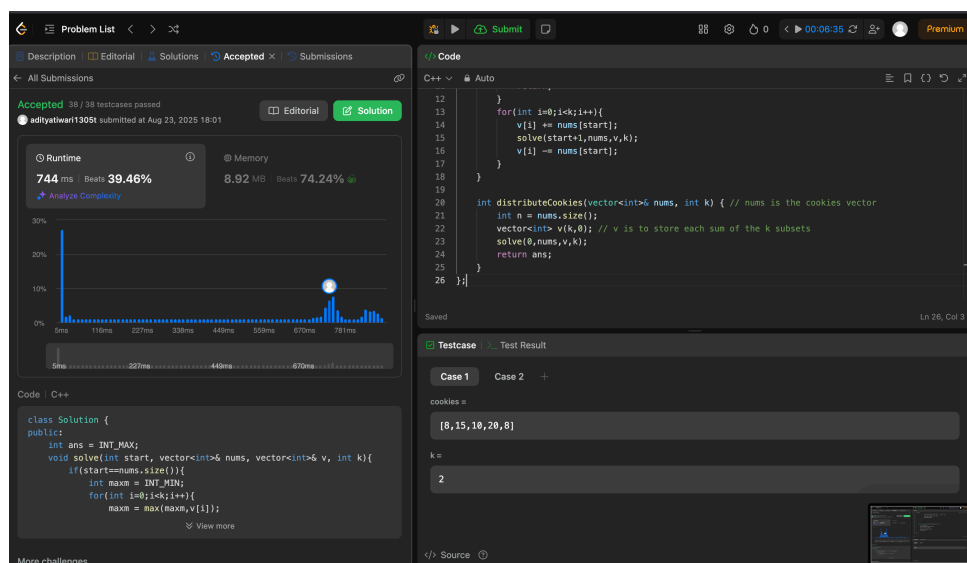
Link: - Palindrome Partitioning - LeetCode



Question 2: Fair Distribution of Cookies

Platform: LeetCode

Link: - Fair Distribution of Cookies - LeetCode



Question 3: N-Queens

Platform: LeetCode

Link: [N-Queens - LeetCode](#)

The screenshot displays the LeetCode interface for the N-Queens problem. The top navigation bar includes 'Problem List', 'Description', 'Editorial', 'Solutions', 'Accepted', and 'Submissions'. The 'Accepted' tab is active, showing a submission by 'adityatiwari13051' at 'Aug 23, 2025 18:01'. The submission status is 'Accepted' with '9 / 9 testcases passed'. The runtime is '0 ms' (Beats 100.00%) and memory is '10.17 MB' (Beats 74.60%). A bar chart shows the runtime distribution. The code is written in C++ and is marked as 'optimized'. The test case input is 'n = 4'. The code implements a recursive backtracking solution for the N-Queens problem.

Runtime: 0 ms | Beats 100.00% | Memory: 10.17 MB | Beats 74.60%

Code:

```
// optimized
class Solution {
public:
    vector<vector<string>> solveNQueens(int n) {
        vector<vector<string>> ans;
        vector<string> board(n);
        string s(n, '.');
        for(int i = 0; i < n; i++){
            board[i] = s;
        }
        vector<int> leftRow(n, UpperDiagonal(2 * n - 1), LowerDiagonal(2 * n - 1));
        helper(0, n, ans, board, leftRow, UpperDiagonal, LowerDiagonal);
        return ans;
    }
};
```

Testcase: Case 1 Case 2 +

n = 4