Name: Anshul

UID: 22BCS16477

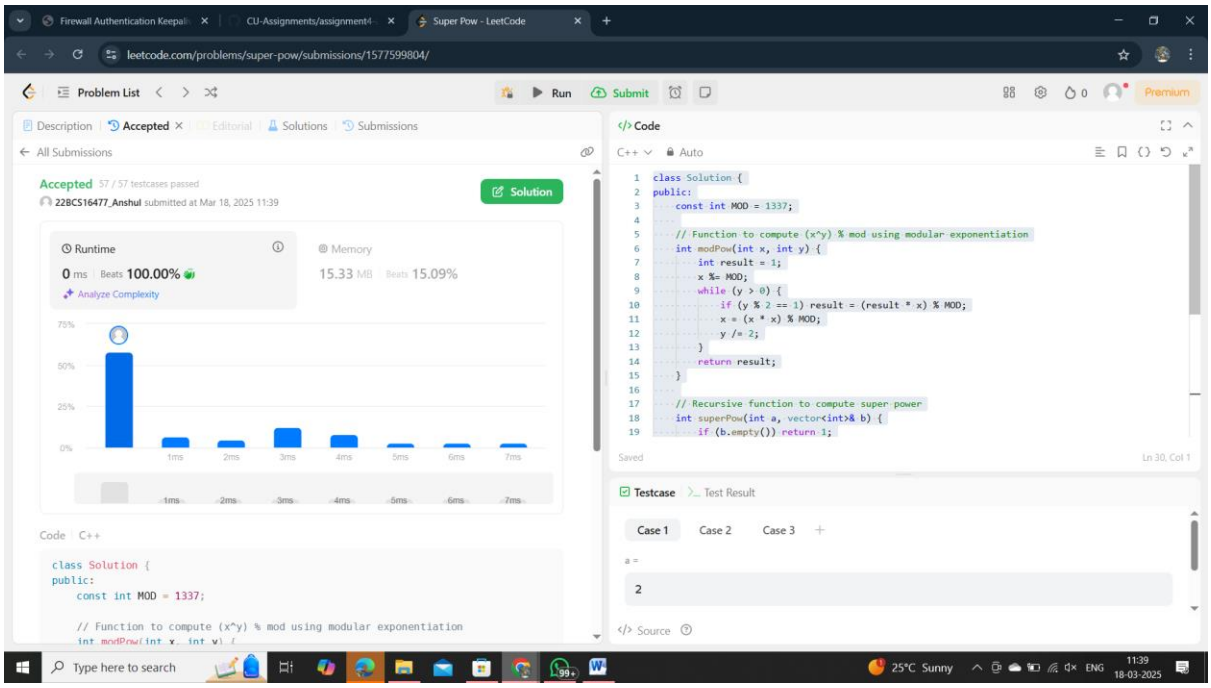
Section/Group: 609(B)

Super Pow

Code:

```
class Solution {
public:
    const int MOD = 1337;

    // Function to compute (x^y) % mod using modular exponentiation
    int modPow(int x, int y) {
        int result = 1;
        x %= MOD;
        while (y > 0) {
            if (y % 2 == 1) result = (result * x) % MOD;
            x = (x * x) % MOD;
            y /= 2;
        }
        return result;
    }

    // Recursive function to compute super power
    int superPow(int a, vector<int>& b) {
        if (b.empty()) return 1;

        int lastDigit = b.back();
        b.pop_back();

        int part1 = modPow(superPow(a, b), 10);  // Recursively compute a^(b/10)
mod 1337
        int part2 = modPow(a, lastDigit);        // Compute a^(b mod 10) mod 1337

        return (part1 * part2) % MOD;
    }
};
```

Output:


Super Pow - LeetCode
leetcode.com/problems/super-pow/submissions/1577599804/
Problem List
Run
Submit
Premium
Description
Accepted
Editorial
Solutions
Submissions
All Submissions
Accepted 57 / 57 testcases passed
22BCS16477_Anshul submitted at Mar 18, 2025 11:39
Solution
Runtime
0 ms | Beats 100.00%
Analyze Complexity
Memory
15.33 MB | Beats 15.09%
Code
C++
class Solution {
public:
const int MOD = 1337;
// Function to compute (x^y) % mod using modular exponentiation
int modPow(int x, int y) {
int result = 1;
x %= MOD;
while (y > 0) {
if (y % 2 == 1) result = (result * x) % MOD;
x = (x * x) % MOD;
y /= 2;
}
return result;
}
// Recursive function to compute super power
int superPow(int a, vector<int>& b) {
if (b.empty()) return 1;
Saved
Ln 30, Col 1
Testcase
Test Result
Case 1
Case 2
Case 3
a =
2
Source
Type here to search
25°C Sunny
ENG
11:39
18-03-2025