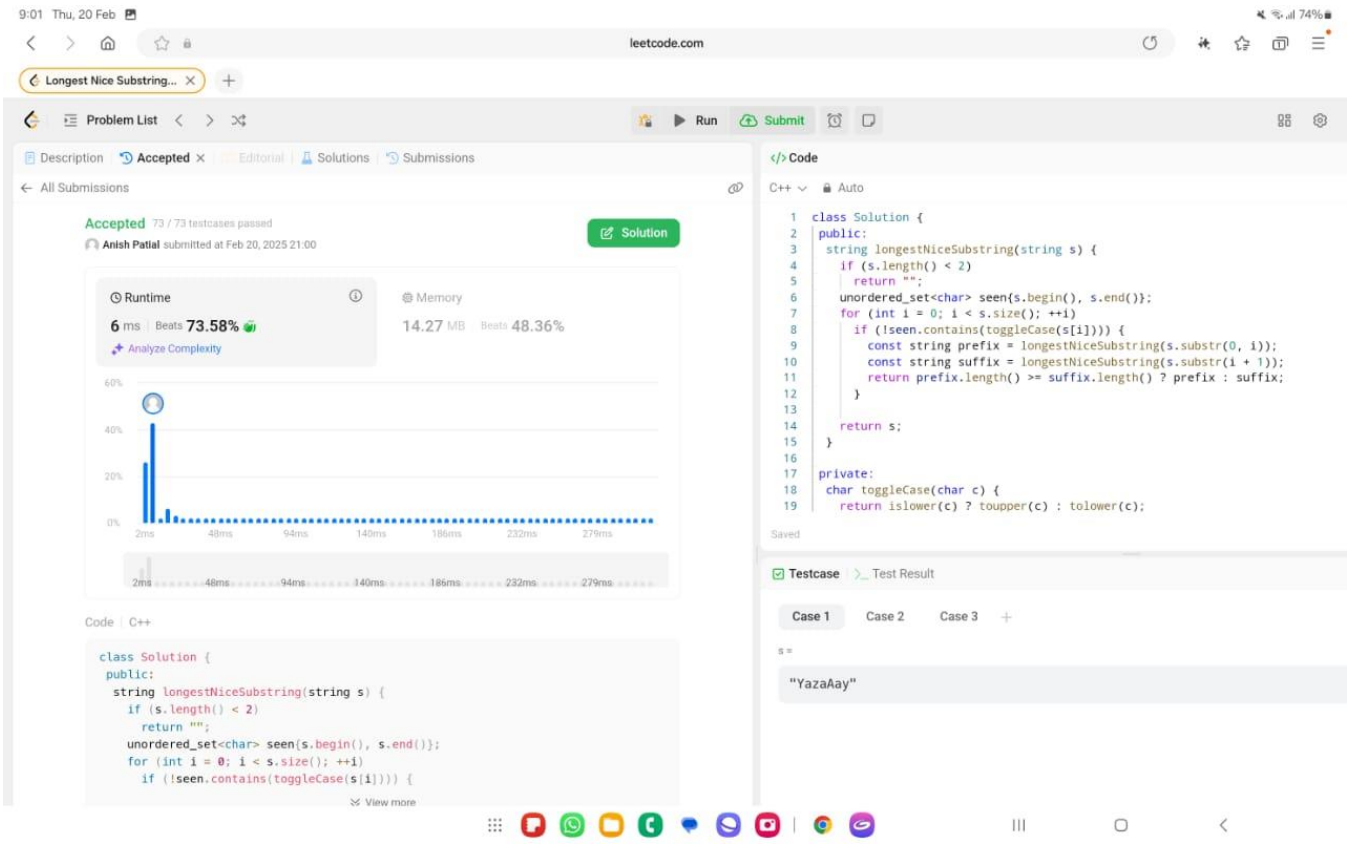
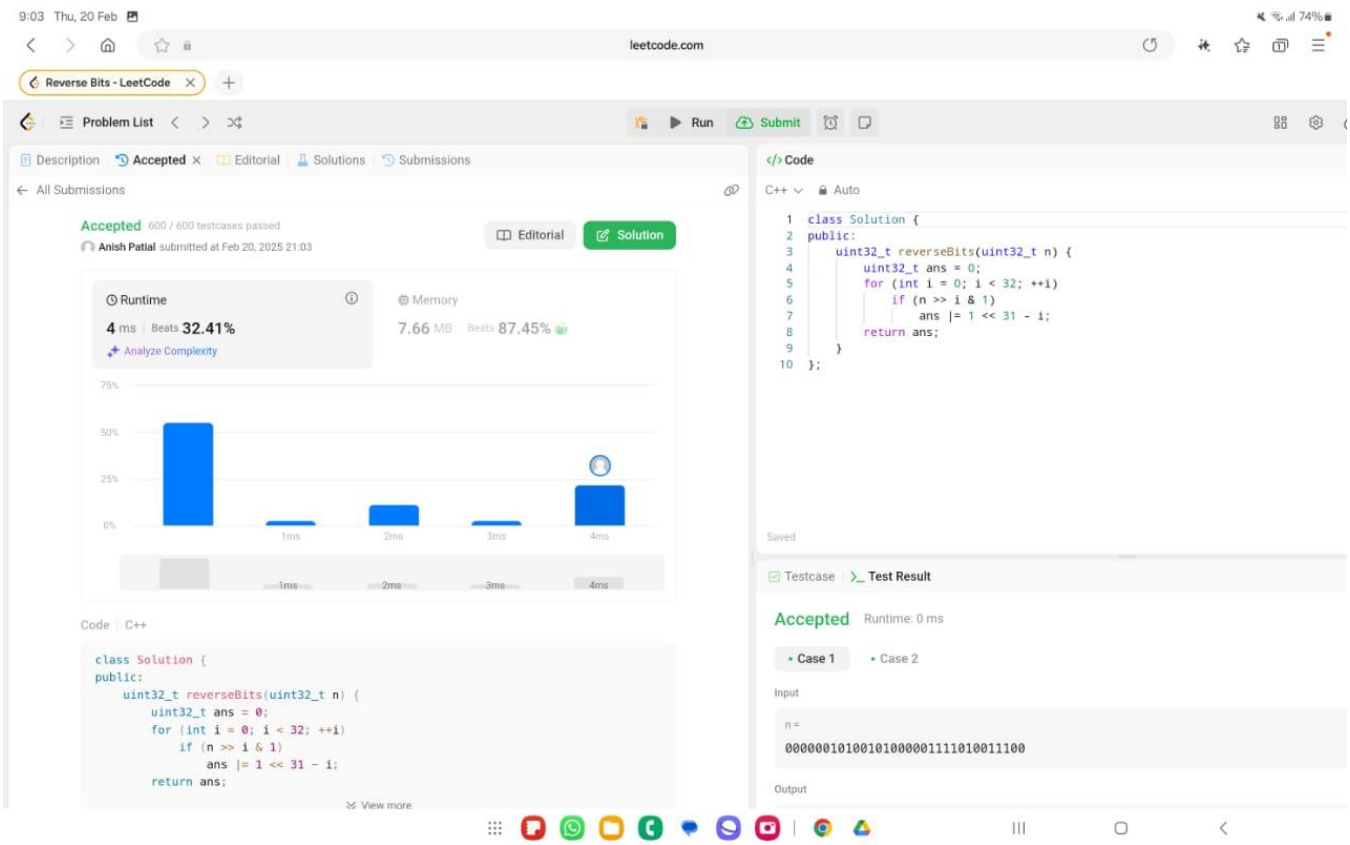
1 Longest Nice Substring

```
class Solution {
 public:
  string longestNiceSubstring(string s) {
    if (s.length() < 2)
      return "";
    unordered_set<char> seen{s.begin(), s.end()};
    for (int i = 0; i < s.size(); ++i)
      if (!seen.contains(toggleCase(s[i]))) {
        const string prefix = longestNiceSubstring(s.substr(0, i));
        const string suffix = longestNiceSubstring(s.substr(i + 1));
        return prefix.length() >= suffix.length() ? prefix : suffix;
      }

    return s;
  }
 private:
  char toggleCase(char c) {
    return islower(c) ? toupper(c) : tolower(c);
  }
};
```


9:01 Thu, 20 Feb
leetcode.com
Longest Nice Substring...
Problem List
Run
Submit
Description
Accepted
Editorial
Solutions
Submissions
All Submissions
Accepted 73 / 73 testcases passed
Anish Patial submitted at Feb 20, 2025 21:00
Solution
Runtime
6 ms | Beats 73.58%
Analyze Complexity
Memory
14.27 MB | Beats 48.36%
60%
40%
20%
0%
2ms
48ms
94ms
140ms
186ms
232ms
279ms
Code | C++
class Solution {
public:
string longestNiceSubstring(string s) {
if (s.length() < 2)
return "";
unordered_set<char> seen{s.begin(), s.end()};
for (int i = 0; i < s.size(); ++i)
if (!seen.contains(toggleCase(s[i]))) {
View more
Code
C++
Auto
class Solution {
public:
string longestNiceSubstring(string s) {
if (s.length() < 2)
return "";
unordered_set<char> seen{s.begin(), s.end()};
for (int i = 0; i < s.size(); ++i)
if (!seen.contains(toggleCase(s[i]))) {
const string prefix = longestNiceSubstring(s.substr(0, i));
const string suffix = longestNiceSubstring(s.substr(i + 1));
return prefix.length() >= suffix.length() ? prefix : suffix;
}
return s;
}
private:
char toggleCase(char c) {
return islower(c) ? toupper(c) : tolower(c);
Saved
Testcase
Test Result
Case 1
Case 2
Case 3
s =
"YazaAay"

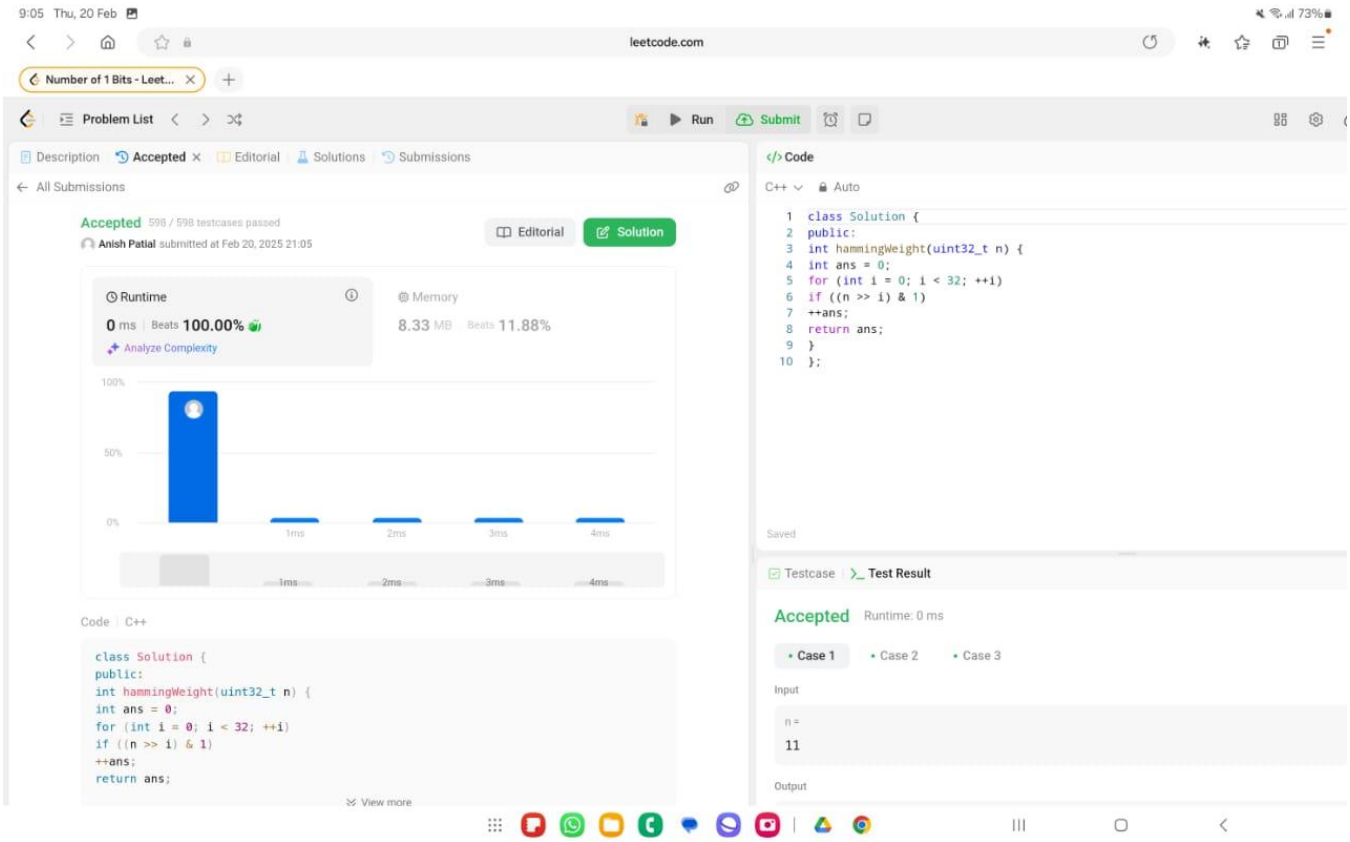
2. Reverse bits

```
class Solution {
 public:
  uint32_t reverseBits(uint32_t n) {
    uint32_t ans = 0;
    for (int i = 0; i < 32; ++i)
      if (n >> i & 1)
        ans |= 1 << 31 - i;
    return ans;
  }
};
```



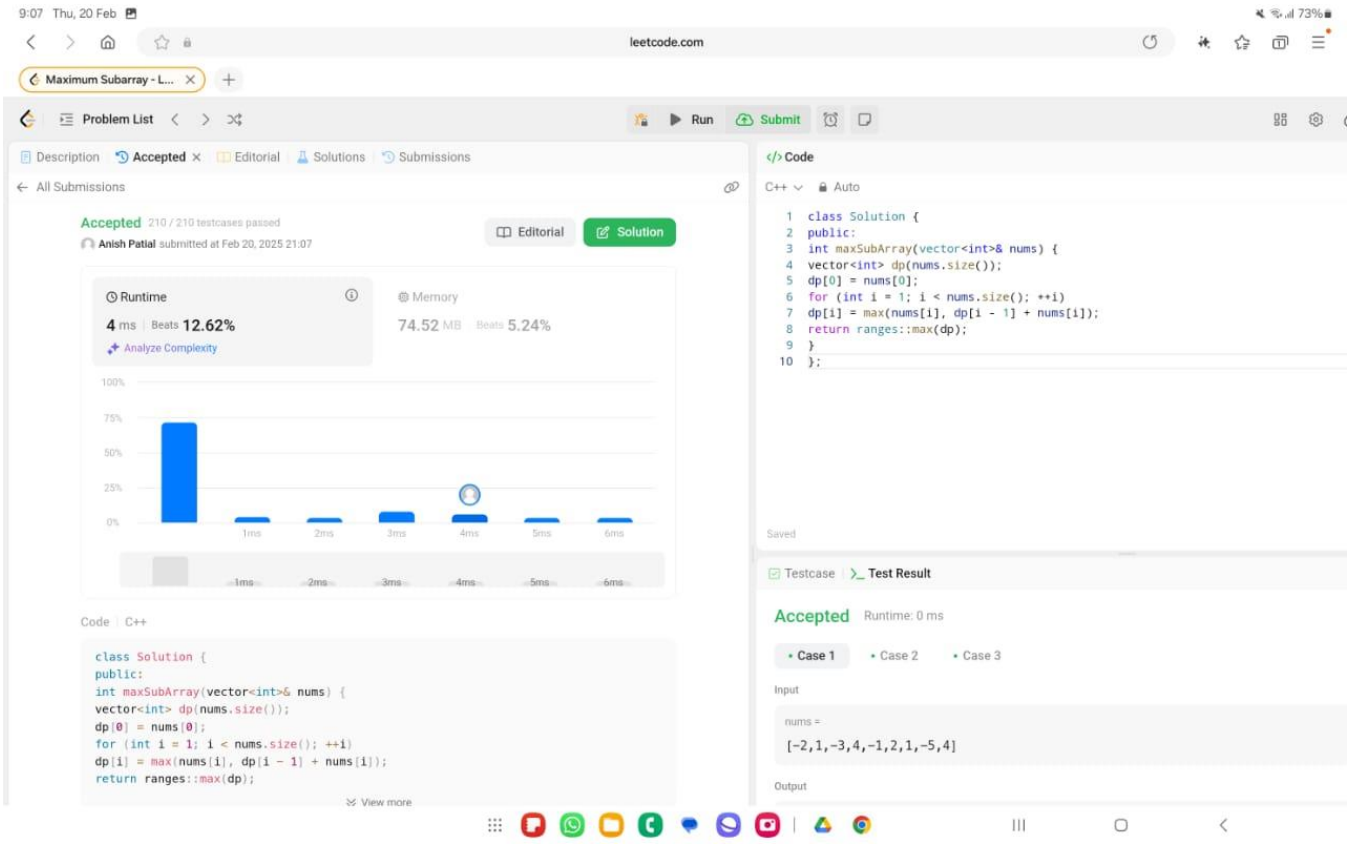
3 Number of 1 bit

```
cclass Solution {
 public:
  int hammingWeight(uint32_t n) {
    int ans = 0;

    for (int i = 0; i < 32; ++i)
      if ((n >> i) & 1)
        ++ans;

    return ans;
  }
};
```



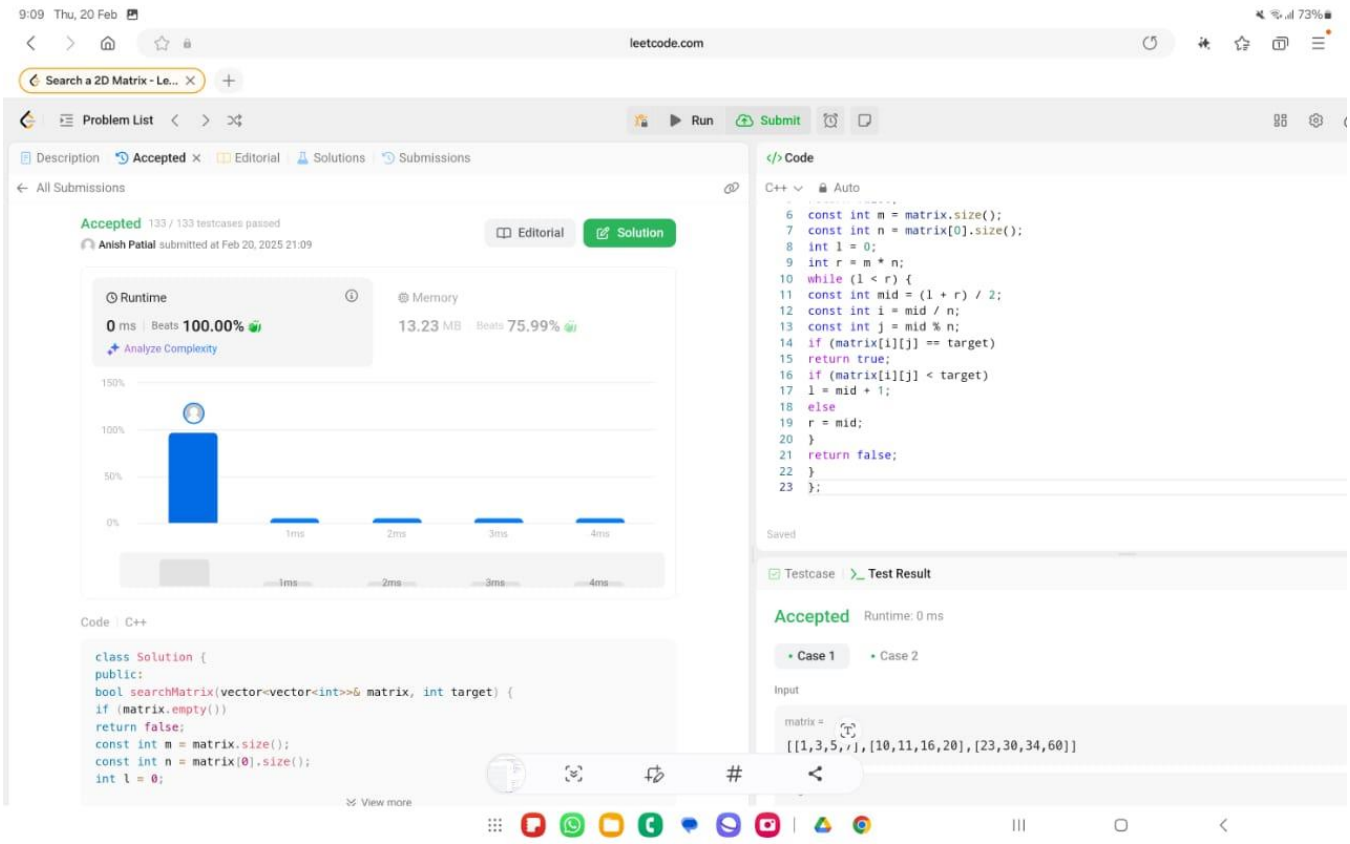
4. maximum Subarray

```
class Solution {
 public:
  int maxSubArray(vector<int>& nums) {
    vector<int> dp(nums.size());
    dp[0] = nums[0];
    for (int i = 1; i < nums.size(); ++i)
      dp[i] = max(nums[i], dp[i - 1] + nums[i]);
    return ranges::max(dp);
  }
};
```



5. Search a 2D Matrix

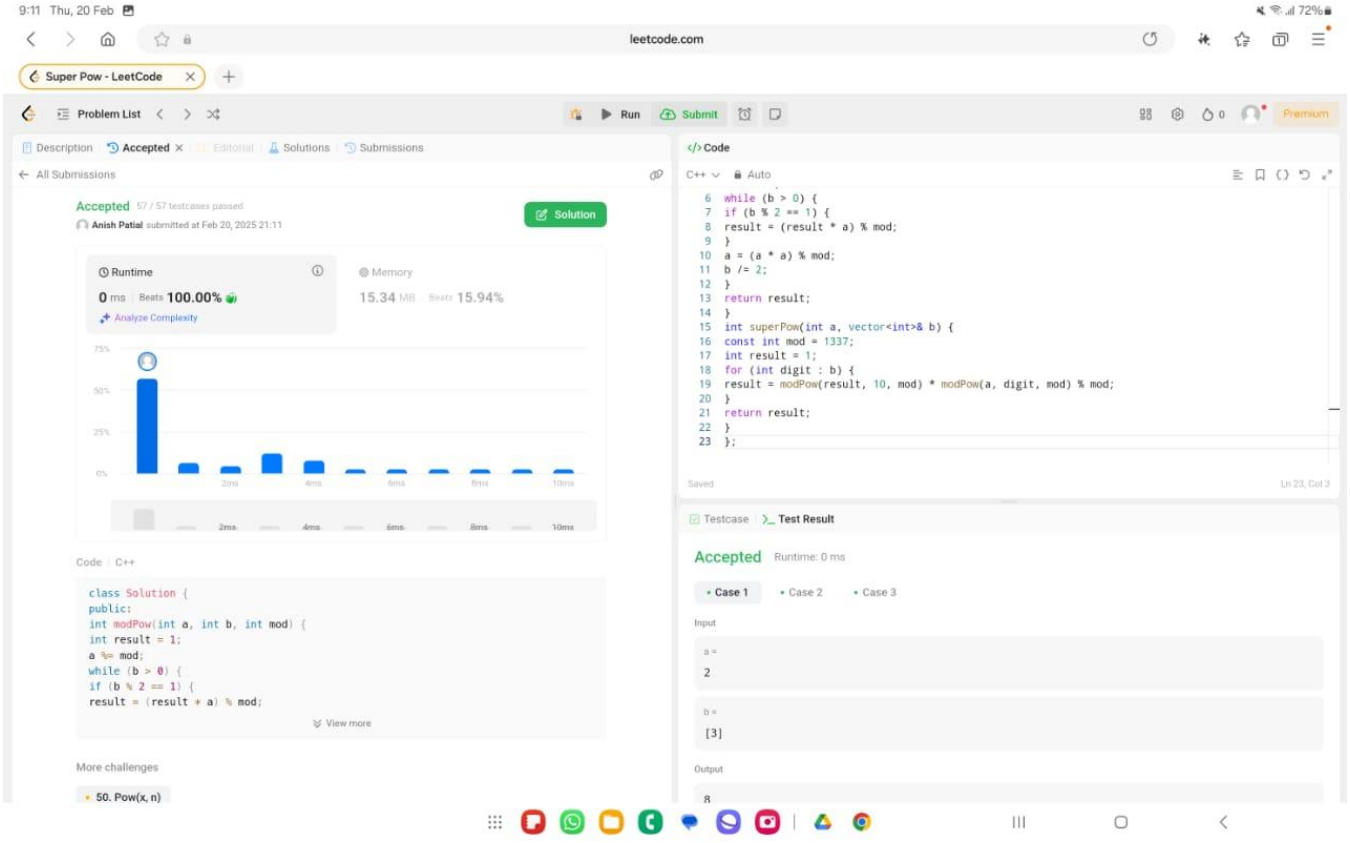
```
class Solution {
 public:
  bool searchMatrix(vector<vector<int>>& matrix, int target) {
    if (matrix.empty())
      return false;
    const int m = matrix.size();
    const int n = matrix[0].size();
    int l = 0;
    int r = m * n;
    while (l < r) {
      const int mid = (l + r) / 2;
      const int i = mid / n;
      const int j = mid % n;
      if (matrix[i][j] == target)
        return true;
      if (matrix[i][j] < target)
        l = mid + 1;
      else
        r = mid;
    }
    return false;
  }
};
```



6. Super Pow

```
class Solution {
public:
    int modPow(int a, int b, int mod) {
    int result = 1;
    a %= mod;
    while (b > 0) {
        if (b % 2 == 1) {
            result = (result * a) % mod;
```

```
        }
        a = (a * a) % mod;
        b /= 2;
    }
    return result;
    }
    int superPow(int a, vector<int>& b) {
    const int mod = 1337;
    int result = 1;
    for (int digit : b) {
        result = modPow(result, 10, mod) * modPow(a, digit, mod) % mod;
    }
    return result;
    }
};
```


9:11 Thu, 20 Feb
72%
leetcode.com
Super Pow - LeetCode
Problem List
Run
Submit
Premium
Description
Accepted
Editorial
Solutions
Submissions
All Submissions
Accepted 57 / 57 testcases passed
Anish Patial submitted at Feb 20, 2025 21:11
Solution
Runtime
0 ms | Beats 100.00%
Analyze Complexity
Memory
15.34 MB | Beats 15.94%
75%
50%
25%
0%
2ms
4ms
6ms
8ms
10ms
Code | C++
class Solution {
public:
int modPow(int a, int b, int mod) {
int result = 1;
a %= mod;
while (b > 0) {
if (b % 2 == 1) {
result = (result * a) % mod;
View more
More challenges
50. Pow(x, n)
Code
C++
Auto
while (b > 0) {
if (b % 2 == 1) {
result = (result * a) % mod;
}
a = (a * a) % mod;
b /= 2;
}
return result;
}
int superPow(int a, vector<int>& b) {
const int mod = 1337;
int result = 1;
for (int digit : b) {
result = modPow(result, 10, mod) * modPow(a, digit, mod) % mod;
}
return result;
}
};
Saved
Ln 23, Col 3
Testcase
Test Result
Accepted
Runtime: 0 ms
Case 1
Case 2
Case 3
Input
a =
2
b =
[3]
Output
8