



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Assignment 6

Student Name: Sagar Kumar Singh

Branch: BE-CSE

Semester: 6th

Subject Name: Advanced Programming

LAB-II

UID: 22BCS13400

Section/Group: 22BCS_IOT-613/B

Subject Code: 22CSP-351

Problem List < > <=>

Run Submit

Description Accepted x Editorial Solutions Subn

All Submissions

Accepted 22 / 22 testcases passed
Sagar Singh submitted at Mar 21, 2025 12:34

Runtime
0 ms | Beats 100.00%
Analyze Complexity

Memory
9.70 MB | Beats 26.86%

Code

```
1 class MyQueue {
2 private:
3     stack<int> s1, s2;
4
5 public:
6     MyQueue() {
7
8     }
9
10    void push(int x) {
11        s1.push(x);
12    }
13 }
```

Testcase | Test Result

Case 1 +

["MyQueue", "push", "push", "peek", "pop", "empty"]

Source

Problem List < > <=>

Run Submit

Description Accepted x Editorial Solutions Subn

All Submissions

Accepted 31 / 31 testcases passed
Sagar Singh submitted at Mar 21, 2025 12:14

Runtime
3 ms | Beats 62.14%
Analyze Complexity

Memory
23.44 MB | Beats 40.94%

Code

```
1 class MinStack {
2 private:
3     stack<int> s, minStack;
4 public:
5     MinStack() {
6
7     }
8
9     void push(int val) {
10        s.push(val);
11        if (minStack.empty() || val <= minStack.top()) {
12            minStack.push(val);
13        }
14    }
15 }
```

Testcase | Test Result

Accepted Runtime: 0 ms

Case 1

Input



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Problem List < > 🔍 Run Submit ⌚ 📄 Premium

Description Accepted × Editorial Solutions < > </> Cod Submit Ctrl Enter

C++ Auto

```
1 class MyCircularQueue {
2 private:
3     vector<int> queue;
4     int head, tail, size, capacity;
5
6 public:
7     MyCircularQueue(int k) {
8         queue.resize(k);
9         head = -1;
10        tail = -1;
11        capacity = k;
12        size = 0;
13    }
14 }
```

Saved Ln 58, Col 1

Testcase > Test Result

Case 1 +

["MyCircularQueue", "enQueue", "enQueue", "enQueue", "enQueue", "Rear", "isFull", "deQueue", "enQueue", "Rear"]

</> Source ?

Accepted 59 / 59 testcases passed

Sagar Singh submitted at Mar 21, 2025 12:31

Runtime

0 ms | Beats 100.00% 🌱

Analyze Complexity

Memory

23.45 MB | Beats 80.26% 🌱

Problem List < > 🔍 Run Submit ⌚ 📄 Premium

Description Editorial Solutions Accepted × Sub: < > </> Code

C++ Auto

```
13 void remove(Node* node) {
14     node->prev->next = node->next;
15     node->next->prev = node->prev;
16 }
17
18 void add(Node* node) {
19     node->next = head->next;
20     node->prev = head;
21     head->next->prev = node;
22     head->next = node;
23 }
24
25 public:
```

Saved Ln 60, Col 1

Testcase > Test Result

Case 1 +

["LRUCache", "put", "put", "get", "put", "get", "put", "get", "get", "get"]

</> Source ?

Accepted 23 / 23 testcases passed

Sagar Singh submitted at Mar 21, 2025 12:32

Runtime

72 ms | Beats 71.72% 🌱

Analyze Complexity

Memory

178.03 MB | Beats 48.28%



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Problem List

Accepted

Editorial

Solutions

Submissions

Accepted 18 / 18 testcases passed

Sagar Singh submitted at Mar 21, 2025 12:28

Runtime

0 ms | Beats 100.00%

Analyze Complexity

Memory

9.51 MB | Beats 18.05%

Code

class MyStack {
private:
queue<int> q;
public:
MyStack() {
}
void push(int x) {
q.push(x);
int n = q.size();
}

Testcase

Test Result

Accepted Runtime: 0 ms

Case 1

Input

Divide and Conquer

Accepted

Editorial

Solutions

Submissions

Accepted 52 / 52 testcases passed

Sagar Singh submitted at Mar 21, 2025 12:33

Runtime

0 ms | Beats 100.00%

Analyze Complexity

Memory

28.09 MB | Beats 84.38%

Code

class Solution {
public:
int majorityElement(vector<int>& nums) {
int count = 0, ele = 0;
for (int num : nums) {
if (count == 0) {
ele = num;
}
if (num == ele) {
count++;
} else {
count--;
}
}

Testcase

Test Result

Accepted Runtime: 0 ms

Case 1 Case 2

Input



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Problem List

Accepted

Editorial

Solutions

Submissions

All Submissions

Accepted 68 / 68 testcases passed

Sagar Singh submitted at Mar 21, 2025 12:35

Solution

Runtime

0 ms | Beats 100.00%

Analyze Complexity

Memory

32.32 MB | Beats 16.28%

100%

75%

50%

25%

C++

Auto

class Solution {
public:
bool isSubPath(ListNode* head, TreeNode* root) {
if (!root) return false;
return dfs(head, root) || isSubPath(head, root->left) || isSubPath(head, root->right);
}

bool dfs(ListNode* head, TreeNode* root) {
if (!head) return true;
if (!root || head->val != root->val) return false;
return dfs(head->next, root->left) || dfs(head->next, root->right);
}
}

Saved

Ln 35, Col 1

Testcase

Test Result

Accepted Runtime: 0 ms

Case 1 Case 2 Case 3

Input

head =