

Name : Sakshi Kapil Badave

PRN : 202401110019

The screenshot shows a C++ IDE with a file named `fact.cpp`. The code defines a class `ArrayMaxMin` with a public method `findMaxMin` that takes an array `arr` and its size `n`. The method checks if `n` is less than or equal to 0, and if so, it prints "Array is empty!". Otherwise, it initializes `max` and `min` with the first element of the array and iterates through the rest of the array to find the maximum and minimum values.

```
1 #include <iostream>
2 using namespace std;
3
4 class ArrayMaxMin {
5 public:
6     void findMaxMin(int arr[], int n) {
7         if (n <= 0) {
8             cout << "Array is empty!" << endl;
9             return;
10        }
11
12        int max = arr[0];
13        int min = arr[0];
14        for (int i = 1; i < n; i++) {
15            if (arr[i] > max)
16                max = arr[i];
17            if (arr[i] < min)
18                min = arr[i];
19        }
20    }
21 };
22
23 int main() {
24     ArrayMaxMin obj;
25     int n;
26     cout << "Enter the number of elements: ";
27     cin >> n;
28     int arr[n];
29     cout << "Enter " << n << " elements: ";
30     for (int i = 0; i < n; i++) {
31         cin >> arr[i];
32     }
33     obj.findMaxMin(arr, n);
34     return 0;
35 }
```

The terminal output shows the program execution:

```
PS C:\Users\Sakshi Badave> cd "c:\college\DSA\"; if ($?) { g++ fact.cpp -o fact }; if ($?) { .\fact }
Enter the number of elements: 6
Enter 6 elements: 10
50
45
67
89
32
Maximum element: 89
Minimum element: 10
PS C:\college\DSA>
```

The screenshot shows a web-based C++ IDE interface. The problem being solved is "Union of Arrays with Duplicates". The code defines a class `Solution` with a public method `findUnion` that takes two vectors `a` and `b` and returns a vector containing the union of the two arrays, with duplicates removed.

```
1 class Solution {
2 public:
3     vector<int> findUnion(vector<int>& a, vector<int>& b) {
4         // code here
5         set<int> s;
6
7         for (int x : a)
8             s.insert(x);
9         for (int x : b)
10            s.insert(x);
11
12        vector<int> unionVec(s.begin(), s.end());
13        return unionVec;
14    }
15 }
```

The output window shows the following results:

- Test Cases Passed: 1111 / 1111
- Attempts: Correct / Total: 1 / 1
- Accuracy: 100%
- Points Scored: 2 / 2
- Time Taken: 0.56
- Your Total Score: 54

The "Solve Next" section lists the following problems:

- Intersection of Arrays with Distinct
- LCM of given array elements
- Perfect Squares in a Range

The "Stay Ahead With:" section is empty.

geeksforgeeks.org/problems/cyclically-rotate-an-array-by-one2614/1

eg

Search...

CoursesTutorialsPracticeJobs

7a🕒🔔S

ProblemEditorialSubmissionsComments

Output Window

Compilation ResultsCustom InputY.O.G.J. (AI Bot)

Problem Solved Successfully✔

Suggest Feedback

Test Cases Passed

1115 / 1115

Attempts : Correct / Total

1 / 1

Accuracy : 100%

Points Scored

1 / 1

Your Total Score: 55 ↑

Time Taken

0.22

Solve Next

Third largest element

Print an array in Pendulum Arrangement

Inverse Permutation

Stay Ahead With:

C++ (17)

Start Timer

```
1 // User function Template for C++
2
3 class Solution {
4 public:
5     void rotate(vector<int> &arr) {
6         // code here
7         int n = arr.size();
8         if (n <= 1)
9             return;
10
11         int last = arr[n - 1];
12
13         for (int i = n - 1; i > 0; i--) {
14             arr[i] = arr[i - 1];
15         }
16
17         arr[0] = last;
18     }
19 };
```

Custom InputCompile & RunSubmit