Search a 2D Matrix (/problems/search-a-2d-matrix/)

Submission Detail

133 / 133 test cases passed.

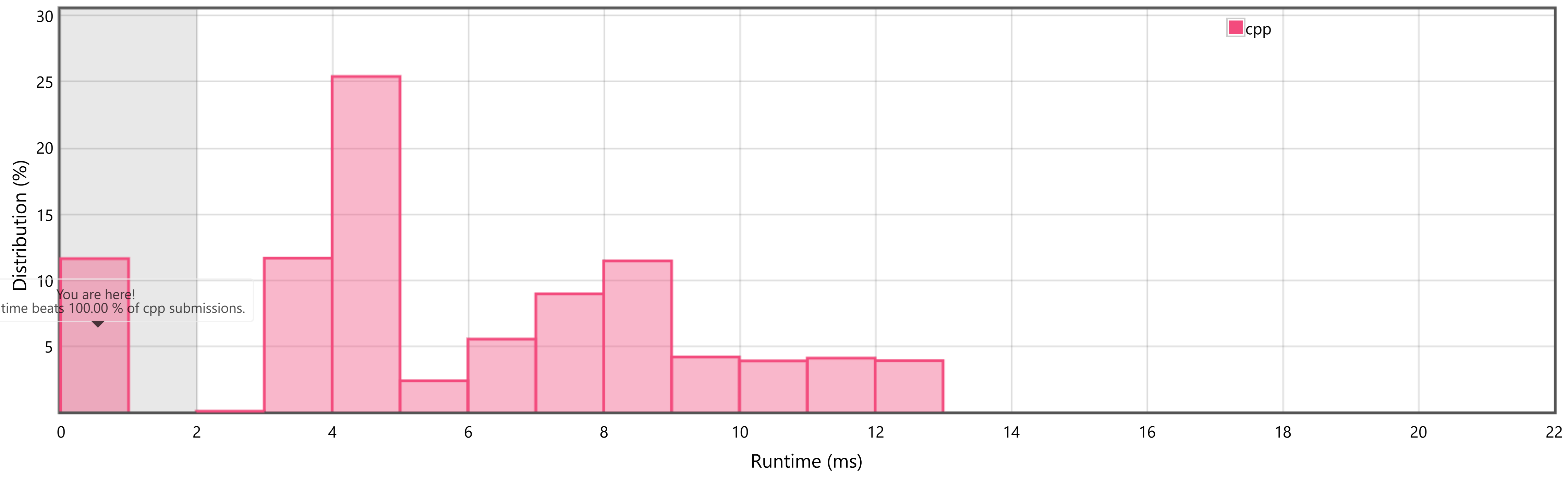
Status: Accepted

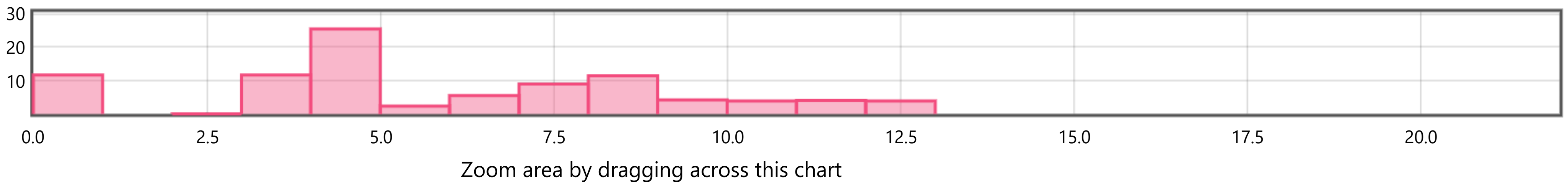
Runtime: **0 ms**

Memory Usage: **9.4 MB**

Submitted: **0 minutes ago**

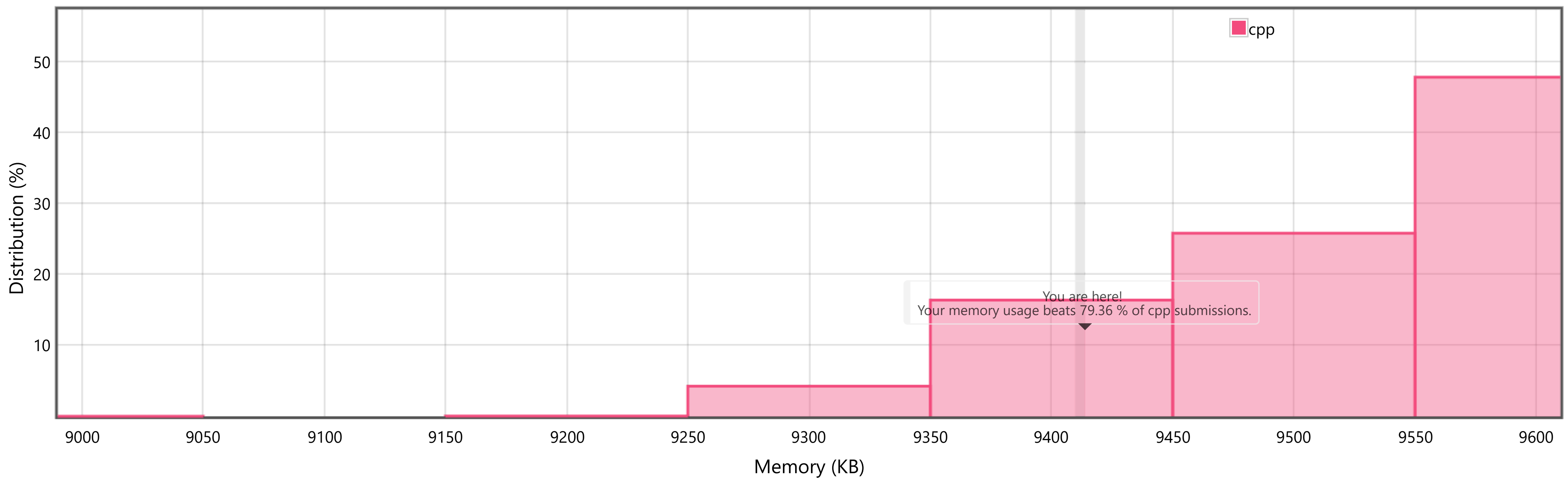
Accepted Solutions Runtime Distribution

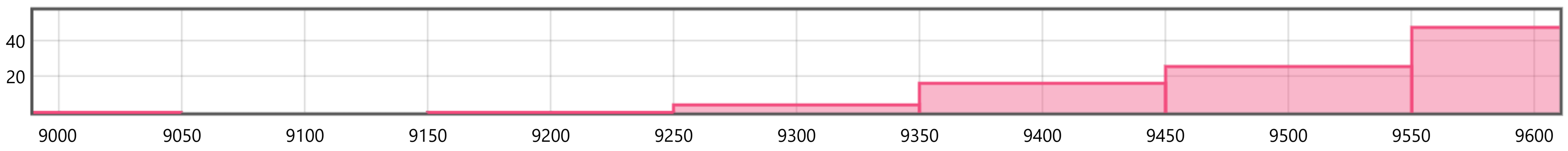




Zoom area by dragging across this chart

Accepted Solutions Memory Distribution





Zoom area by dragging across this chart

Invite friends to challenge **Search a 2D Matrix**

25

Submitted Code: 0 minutes ago

Language: cpp

Edit Code

```
class Solution {
public:
    int search(vector<int>& arr, int target){
        int l = 0;
        int r = arr.size() - 1;
        int mid = l + (r - l) / 2;
        while(l <= r){
            mid = l + (r - l) / 2;
            if(arr[mid] == target)
                return mid;
            else if(target > arr[mid])
                l = mid + 1;
            else r = mid - 1;
        }
        return -1;
    }
    bool searchMatrix(vector<vector<int>>& matrix, int target) {
        /*
            Approach->
                Traverse every row
                    check if target lies between first and last element of row
                    if(lies)
                        then do binary search in current row and find if target is present in current row
                    else
                        continue to next row
        */
        for(int i = 0; i < matrix.size(); i++){
            if(target >= matrix[i][0] and target <= matrix[i][matrix[i].size() - 1]){
                int ans = search(matrix[i], target);
                if(ans != -1)
                    return true;
                else return false;
            }
        }
        return false;
    }
};
```

Back to problem (/problems/search-a-2d-matrix/)

Help Center (/support) | Jobs (/jobs) | Bug Bounty (/bugbounty) | Online Interview (/interview/) | Students (/student) | Terms (/terms) | Privacy Policy (/privacy)

United States (/region)