DSA PRACTICE QUESTIONS

18/11/2024

1.BUBBLE SORT: TC: O(N)


```java
 9  // User function Template for Java
10
11  class Solution {
12      public static void bubbleSort(int arr[]) {
13          int n = arr.length;
14          for (int i = 0; i < n - 1; i++) {
15              boolean swap = false;
16              for (int j = 0; j < n - i - 1; j++) {
17                  if (arr[j] > arr[j + 1]) {
18                      int temp = arr[j];
19                      arr[j] = arr[j + 1];
20                      arr[j + 1] = temp;
21                      swap = true;
22                  }
23              }
24              if (!swap) break;
25          }
26      }
27  }
28
29  // } Driver Code Ends
```

Output Window

Compilation Results   Custom Input   Y.O.G.I. (AI Bot)

Problem Solved Successfully ✓   Suggest Feedback

Test Cases Passed

1115 / 1115

Attempts : Correct / Total

1/1

Accuracy : 100%

Points Scored ⓘ

2 / 2

Your Total Score: 102 ↑

Time Taken

0.58

2.QUICK SORT: TC: O(NLOGN)


```java
31
32  class Solution {
33      public static int partition(int[] arr, int low, int high) {
34          int pivot = arr[high];
35          int i = low - 1;
36          for (int j = low; j < high; j++) {
37              if (arr[j] <= pivot) {
38                  i++;
39                  int temp = arr[i];
40                  arr[i] = arr[j];
41                  arr[j] = temp;
42              }
43          }
44          int temp = arr[i + 1];
45          arr[i + 1] = arr[high];
46          arr[high] = temp;
47          return i + 1;
48      }
49
50      public static void quickSort(int[] arr, int low, int high) {
51          if (low < high) {
52              int p = partition(arr, low, high);
53              quickSort(arr, low, p - 1);
54              quickSort(arr, p + 1, high);
55          }
56      }
57
58      public static void main(String[] args) {
59          int[] arr = {4, 1, 3, 9, 7};
60          quickSort(arr, 0, arr.length - 1);
61          for (int num : arr) {
62              System.out.print(num + " ");
63          }
64      }
65  }
```

Output Window

Compilation Results   Custom Input   Y.O.G.I. (AI Bot)

Problem Solved Successfully ✓   Suggest Feedback

Test Cases Passed

1120 / 1120

Attempts : Correct / Total

1/1

Accuracy : 100%

Points Scored ⓘ

4 / 4

Your Total Score: 106 ↑

Time Taken

0.6

Solve Next

Merge Sort   Maximum Intervals Overlap   Print the Sequence

Kick start your career with GfG 160!

3.NON REPEATING CHARACTER: TC:O(N)

Problem Solved Successfully ✓

Suggest Feedback

Test Cases Passed

1130 / 1130

Attempts : Correct / Total

1/1

Accuracy : 100%

Points Scored ⓘ

2/2

Time Taken

0.37

Your Total Score: 108 ↑

```java
class Solution {
    static char nonRepeatingChar(String s) {
        int[] freq = new int[26];
        for (int i = 0; i < s.length(); i++) {
            freq[s.charAt(i) - 'a']++;
        }
        for (int i = 0; i < s.length(); i++) {
            if (freq[s.charAt(i) - 'a'] == 1) {
                return s.charAt(i);
            }
        }
        return '$';
    }
}
```

5. PRINT LARGEST NUMBER: TC:O(N)

Compilation Results   Custom Input   Y.O.G.I. (AI Bot)

Problem Solved Successfully ✓

Suggest Feedback

Test Cases Passed

1111 / 1111

Attempts : Correct / Total

1/1

Accuracy : 100%

Points Scored ⓘ

4/4

Time Taken

1.46

Your Total Score: 112 ↑

```java
// User function Template for Java
class Solution {
    String printLargest(int[] arr) {
        String[] nums = new String[arr.length];
        for (int i = 0; i < arr.length; i++) {
            nums[i] = String.valueOf(arr[i]);
        }

        Arrays.sort(nums, (a, b) -> (b + a).compareTo(a + b));

        if (nums[0].equals("0")) {
            return "0";
        }

        StringBuilder result = new StringBuilder();
        for (String num : nums) {
            result.append(num);
        }

        return result.toString();
    }
}
```

6.K LARGEST ELEMENT: TC: O(NLOGN)

Problem Solved Successfully ✅

Test Cases Passed

1111 / 1111

Attempts : Correct / Total

1/1

Accuracy : 100%

Points Scored ℹ️

4 / 4

Your Total Score: 116 ↑

Time Taken

0.09

```python
class Solution:
    def kLargest(self, arr, k):
        le = len(arr)
        l = []
        arr.sort()
        for i in range(k):
            m = arr[le-1-i]
            l.append(m)
        return l

# } Driver Code Ends
```