

# EXPERIMENT FOR PROGRAMMING ABILITY AND LOGIC BUILDING-1 (JAVA)

## Pdf-2

### EXPERIMENT-1

The screenshot shows a web-based IDE interface. On the left, the 'Output Window' is open, displaying 'Compilation Results' for 'Custom Input' by 'Y.O.G.I. (AI Bot)'. It states 'Problem Solved Successfully' with a green checkmark. Below this, it shows 'Test Cases Passed: 1121 / 1121', 'Attempts: Correct / Total: 4 / 4', and 'Accuracy: 100%'. The 'Time Taken' is '0.78'. On the right, the code editor shows a Java solution for finding the kth smallest element in an array. The code is as follows:

```
1 class Solution {
2     public int kthSmallest(int[] arr, int k) {
3         // Code here
4         Arrays.sort(arr);
5         return arr[k-1];
6     }
7 }
8
```

At the bottom right, there are buttons for 'Custom Input', 'Compile & Run', and 'Submit'.

### EXPERIMENT-4

The screenshot shows a web browser displaying a LeetCode submission for the 'Find the Duplicate Number' problem. The URL is 'leetcode.com/problems/find-the-duplicate-number/submissions/1915343244/'. The submission is 'Accepted' with 59/59 testcases passed, submitted by 'Shrutisinh\_07' on Feb 11, 2026, 09:46. The 'Runtime' is 39 ms, beating 6.52% of solutions. The 'Memory' is 79.36 MB, beating 91.60% of solutions. A bar chart shows the distribution of runtime and memory usage. The 'Code' tab is active, showing a Java solution for finding the duplicate number in an array. The code is as follows:

```
1 class Solution {
2     public int findDuplicate(int[] nums) {
3         Arrays.sort(nums);
4
5         for(int i = 0; i < nums.length - 1; i++) {
6             if(nums[i] == nums[i + 1]) {
7                 return nums[i];
8             }
9         }
10    }
11 }
```

The 'Test Result' tab shows 'Accepted' with a runtime of 1 ms. It lists three test cases: Case 1, Case 2, and Case 3, all of which are passed. The input for Case 1 is 'nums = [1,3,4,2,2]' and the output is '1'.

## EXPERIMENT-5

The screenshot shows the GeeksforGeeks website interface for the 'Merge Without Extra Space' problem. The browser tabs include 'JAVA EXP-2 - Google Docs', 'Find the Duplicate Number - Le...', and 'Merge Without Extra Space | P...'. The URL bar shows 'geeksforgeeks.org/problems/merge-two-sorted-arrays-1587115620/1'. The page header features the GeeksforGeeks logo, a search bar, and navigation links for 'Courses', 'Tutorials', 'Practice', and 'Jobs'. The main content area is divided into two panels. The left panel, titled 'Output Window', displays 'Compilation Results' and a 'Problem Solved Successfully' message with a green checkmark. It includes statistics: 'Test Cases Passed: 1111 / 1111', 'Attempts: Correct / Total: 1 / 1', 'Accuracy: 100%', 'Points Scored: 4 / 4', and 'Time Taken: 1.05'. The right panel shows the Java code for the solution, which is a class 'Solution' with a method 'mergeArrays' that merges two sorted arrays without extra space. The code is as follows:

```
1 class Solution {
2     public void mergeArrays(int a[], int b[]) {
3         // code here
4
5         int n = a.length;
6         int m = b.length;
7         for(int i = 0; i < n; i++) {
8             if(a[i] > b[0]) {
9
10                int temp = a[i];
11                a[i] = b[0];
12                b[0] = temp;
13
14                int first = b[0];
15                int k;
16
17                for(k = 1; k < m && b[k] < first; k++) {
18                    b[k-1] = b[k];
19                }
20                b[k-1] = first;
21            }
22        }
23    }
24 }
25 }
```

At the bottom right, there are buttons for 'Custom Input', 'Compile & Run', and 'Submit', along with a 'Ctrl + Enter' shortcut indicator.

## EXPERIMENT-6