

NAME: GAIRE ANANTA PRASAD

ID: M24W0272

## 1. CODE FOR BUBBLE SORT

```
bisectionMethod.java U newtonsMethod.java U BubbleSortExperiment.java U X BubbleSort.java U
BubbleSortExperiment.java > BubbleSortExperiment > bubbleSort(int[])
1  /**
2   * NAME: GAIRE ANANTA PRASAD
3   * ID: M24W0272
4   */
5
6  public class BubbleSortExperiment {
7
8      /**
9       * Function to implement Bubble Sort algorithm.
10      *
11      * @param arr the array to be sorted
12      */
13      public static void bubbleSort(int[] arr) {
14          int n = arr.length;
15          int comparisons = 0; // Variable to count the number of comparisons
16
17          for (int i = 0; i < n - 1; i++) {
18              // Track if any swapping occurred to optimize the sorting process
19              boolean swapped = false;
20
21              for (int j = 0; j < n - i - 1; j++) {
22                  comparisons++; // Increment comparisons count
23
24                  // Swap if the current element is greater than the next element
25                  if (arr[j] > arr[j + 1]) {
26                      int temp = arr[j];
27                      arr[j] = arr[j + 1];
28                      arr[j + 1] = temp;
29                      swapped = true;
30                  }
31              }
32
33              // If no elements were swapped, the array is already sorted
34              if (!swapped) break;
35          }
36
37          // Print the sorted array and the number of comparisons
38
39          System.out.println("Sorted array:");
40          for (int i : arr) {
41              System.out.print(i + " ");
42          }
43          System.out.println("\nNumber of comparisons: " + comparisons);
44
45          /**
46           * Function to print the array.
47           *
48           * @param arr the array to be printed
49           */
50          public static void printArray(int[] arr) {
51              for (int i : arr) {
52                  System.out.print(i + " ");
53              }
54              System.out.println();
55          }
56
57          Run | Debug
58          public static void main(String[] args) {
59              // Experiment 1: Small data size
60              int[] smallData = { 5, 1, 4, 2, 8 };
61              System.out.println("Experiment 1: Small Data Size");
62              System.out.println("Original array:");
63              printArray(smallData);
64              bubbleSort(smallData);
65
66              // Experiment 2: Large data size
67          }
68      }
69  }
```

GAIRE ANANTA PRASAD

M24W0272

```

BubbleSortExperiment.java U
newtonsMethod.java U
BubbleSortExperiment.java U X
BubbleSort.java U

BubbleSortExperiment.java > ...
6 public class BubbleSortExperiment {
50     public static void printArray(int[] arr) {
53         }
54         System.out.println();
55     }
56
57     Run | Debug
58     public static void main(String[] args) {
59         // Experiment 1: Small data size
60         int[] smallData = { 5, 1, 4, 2, 8 };
61         System.out.println(x:"Experiment 1: Small Data Size");
62         System.out.println(x:"Original array:");
63         printArray(smallData);
64         bubbleSort(smallData);
65
66         // Experiment 2: Large data size
67         int[] largeData = { 64, 34, 25, 12, 22, 11, 90, 78, 45, 67, 89, 10, 5, 3, 2, 1 };
68         System.out.println(x:"\nExperiment 2: Large Data Size");
69         System.out.println(x:"Original array:");
70         printArray(largeData);
71         bubbleSort(largeData);
72     }
73 }

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS GITLENS
Experiment 1: Small Data Size
Original array:
5 1 4 2 8
Sorted array:
1 2 4 5 8
Number of comparisons: 9

Experiment 2: Large Data Size
Original array:
64 34 25 12 22 11 90 78 45 67 89 10 5 3 2 1
Sorted array:
1 2 3 5 10 11 12 22 25 34 45 64 67 78 89 90
Number of comparisons: 120

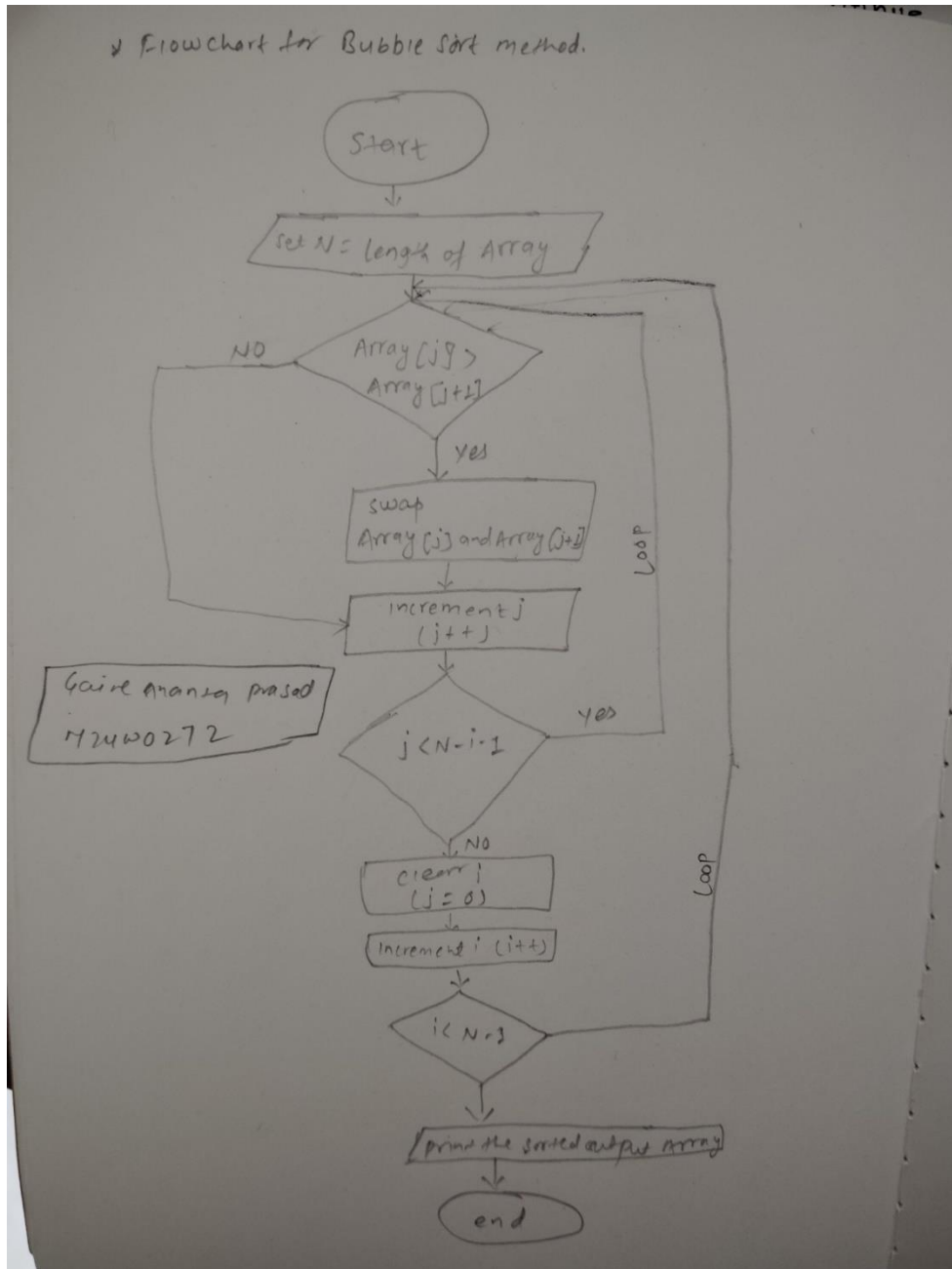
pad 0 0 0 %0 0 Live Share Java: Ready Ln 74, Col 1 Spaces: 4 UTF-8 CRLF {} Java Go Live Quokka

```

← Output

GAIRE ANANTA PRASAD  
M24W0272

## 2. FLOW CHART FOR BUBBLE



### 3. PSEUDOCODE FOR BUBBLE SORT

*tr*

*\* Pseudocode for Bubble Sort Algorithm:*

*Start!*

- 1 Initialization: Read the array of given items from the user.*
- 2 Compute: Take the first element (index = 0)*
- 3 Compute: Compare the current element with the next element.*
  - iterate 'j' from 0 to 'n-i-1'*
  - compare arr[j] with arr[j+1]*
  - If arr[j] > arr[j+1]*
  - 4 swap arr[j] and arr[j+1]*
- 5 Compute: else, if the current element is less than the next element, then move to the next element.*
- 6 - Repeat step 3 to step 5 until all elements are sorted.*

*Output: The array will be sorted in ascending order.*

*End the program.*

Gaire Ananta Prasad  
M24W0272