NIGER DELTA UNIVERSITY
WILBERFORCE ISLAND, AMASSOMA
BAYELSA STATE

FACULTY OF SCIENCE
DEPARTMENT OF COMPUTER SCIENCE

ASSIGNMENT II

COURSE TITLE: ALGORITHM

COURSE CODE: CMP 421

COURSE LECTURER: DR KENEKAYORO

NAME:
FRANCIS EBUKA PROGRESS

MATRIC NO.:
UG/17/1434

Bubble Sort in java without stopping

```
class BubbleSort
{
   void bubbleSort(int arr[])
   {
      int n = arr.length;
      for (int i = 0; i < n-1; i++)
         for (int j = 0; j < n-i-1; j++)
            if (arr[j] > arr[j+1])
            {
               // swap arr[j+1] and arr[j]
               int temp = arr[j];
               arr[j] = arr[j+1];
               arr[j+1] = temp;
            }
   }
   /* Prints the array */
   void printArray(int arr[])
   {
      int n = arr.length;
      for (int i=0; i<n; ++i)
         System.out.print(arr[i] + " ");
      System.out.println();
   }
   // Driver method to test above
   public static void main(String args[])
{
      BubbleSort ob = new BubbleSort();
      int arr[] = {64, 34, 25, 12, 22, 11, 90};
      ob.bubbleSort(arr);
      System.out.println("Sorted array");
      ob.printArray(arr);
   }
}
```

or

Bubble sort with stopping

```
class BubbleSort {
  void bubbleSort(int arr[]) {   //sorting method
    int size = arr.length;
    for (int i = 0; i < size - 1; i++) {
      boolean flag = true;
      for (int j = 0; j < size - i - 1; j++) {
        if (arr[j] > arr[j + 1]) {
          int temp = arr[j];
          arr[j] = arr[j + 1];
          arr[j + 1] = temp;
          flag = false;
        }
      }
      if (flag == true)
        break;
```

```
  }
 }
 void display(int arr[]) {   //method for displaying the elements
    int size = arr.length;
            for (int i = 0; i < size; i++)
                    System.out.println(arr[i]+" ");
  }
 public static void main(String args[]) {   //main method or driver method
   int[] arr = { -2, 45, 0, 11, -9 };
   BubbleSort  bs = new BubbleSort();
   System.out.println("Elements before Sorting:");
   bs.display(arr);
   bs.bubbleSort(arr);
   System.out.println("Elements after Sorting:");
   bs.display(arr);
 }
}
```

Insertion Sort in java

```
import java.util.*;
class InsertionSort {
   //method for sorting the elements
   void insertionSort(int arr[]) {
      int size = arr.length;
      for (int i = 1; i < size; i++) {
         int tmp = arr[i];
         int j = i - 1;
         while (j >= 0 && tmp < arr[j]) {
            arr[j + 1] = arr[j];
            --j;
            }
            arr[j + 1] = tmp;
            }
            }
            // method for printing the elements
            void display(int arr[]) {
               int size = arr.length;
               for (int i = 0; i < size; i++)
               System.out.print(arr[i]+" ");
               System.out.println();

            } // Main method or driver method
            public static void main(String args[]) {
               int[] arr = { 9, 5, 1, 4, 3 };
               InsertionSort  ob = new InsertionSort();
               System.out.println("Elements before sorting: ");
               ob.display(arr);
               ob.insertionSort(arr);
               System.out.println("Elements after sorting: ");
               ob.display(arr);
               }
```

```
}
//Output of the program: Elements before sorting: 9 5 1 4 3 Elements after sorting: 1 3 4 5 9
```