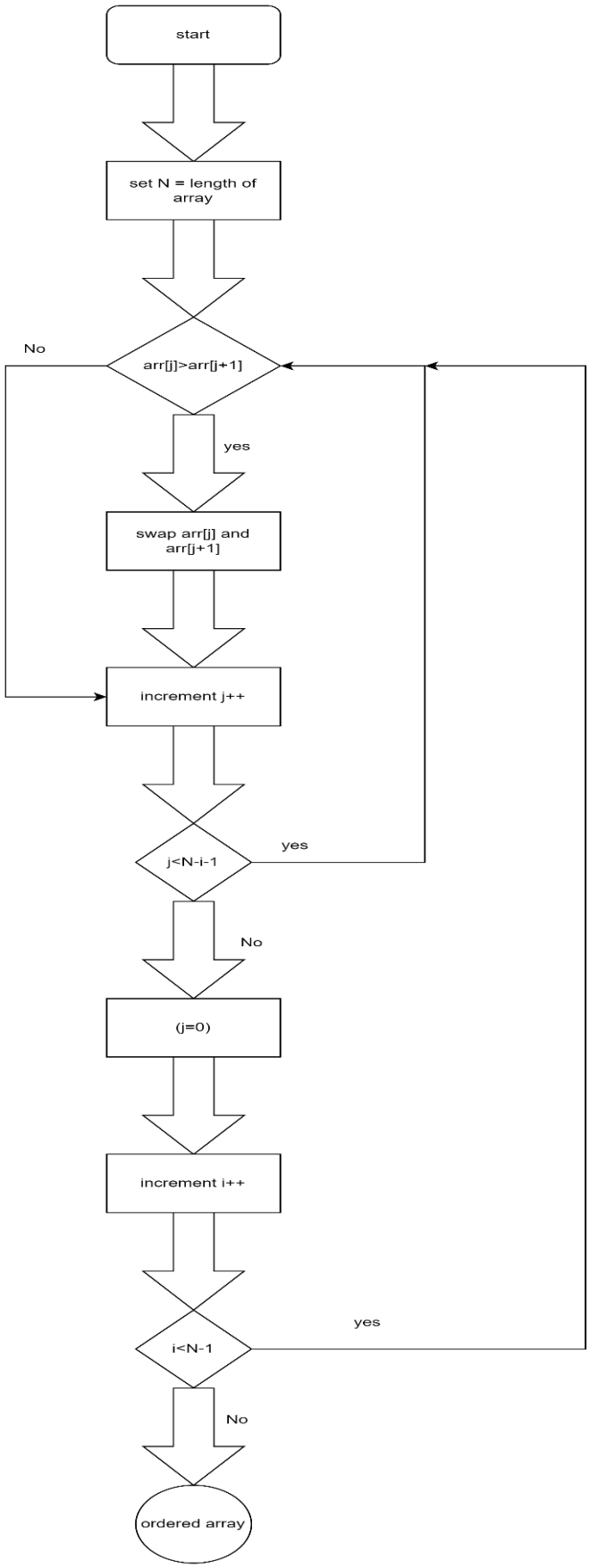
Assignment 1:

Pseudocode and Flowchart for Sorting Algorithm – Write pseudocode and create a flowchart for a bubble sort algorithm. Provide a brief explanation of how the algorithm works and a simple array of integers to demonstrate a dry run of your algorithm.

Pseudocode

```
Bubble Sort( x : list of sortable items)
Start
   n := length(x)
   repeat
      swapped := false
      for i := 1 to (n – 1)
         if x[i-1] > x[i] then
            swap(x[i-1], x[i])
            swapped := true
         end if
      end for
      n := n - 1
   until not swapped
End.
```

Flowchart :


start
set N = length of array
arr[j]>arr[j+1]
No
yes
swap arr[j] and arr[j+1]
increment j++
j<N-i-1
yes
No
(j=0)
increment i++
i<N-1
yes
No
ordered array

CODE :

```
#include <stdio.h>
void bubbleSort(int arr[], int n) {
    int i, j, temp;
    for (i = 0; i < n - 1; i++) {
            for (j = 0; j < n - i - 1; j++) {
          if (arr[j] > arr[j + 1]) {
              temp = arr[j];
              arr[j] = arr[j + 1];
              arr[j + 1] = temp;
          }
       }
    }
}
void printArray(int arr[], int size) {
    int i;
    for (i = 0; i < size; i++)
       printf("%d ", arr[i]);
    printf("\n");
}
int main() {
    int arr[] = {64, 34, 25, 12, 22, 11, 90};
    int n = sizeof(arr) / sizeof(arr[0]);
    printf("Original array: \n");
    printArray(arr, n);
    bubbleSort(arr, n);
    printf("Sorted array: \n");
    printArray(arr, n);
    return 0;
}
```

Algorithm :

1. Start
2. Initialize n to length of array.
3. Set swapped flag to false.
4. Repeat until swapped is false.
5. Reset swapped flag to false.
6. Loop through array elements
7. Compare current element with next element
8. If current element > next element then Swap elements
9. Set swapped flag to true
10. Decrement n until swapped is false.
11. End.