**ASSIGNMENT -1**

**1.Find the maximum length of an array.**

**PSEUDO CODE:**

|  |
| --- |
| **Int[] a**  **Int max=a[0];**  **Int i=1;**  **while(i<a.length)**  **{**  **If(a[i]>max)**  **[**  **max=a[i];**  **}**  **i++;**  **}** |

**Pre- Condition**

* Variables **‘I’** that acts as a loop counter.
* Variable **max** stores the maximum value of an array.
* Initialise **max= a[0]** and **i=1**.

**Post-Condition**

* Value of the maximum must be equal to the maximum value of an array **max=a[i].**

**Loop Variant**

* The loop condition **i<a.length** ensures that the loop stop when the ‘i’ is equal to the size of an array.

**Loop invariant**

* Compare max with new value a[i] and found the largest element and stored in max variable **(a[i]>max).**

**2. Moves zeros to the end of array.**

**PSEUDO CODE:**

|  |
| --- |
| **moveZerosToEnd(int[] a{**  **int left = 0;**  **int right = 0;**    **while (right < a.length) {**  **if (a[right] != 0) {**  **int temp = a[left];**  **a[left] = a[right];**  **a[right] = temp;**  **left++;**  **}**  **right++;**  **}**  **}** |

**Pre-Condition**

* initialize the two pointers **left =0** and **right=0** to point to the beginning of the array.

**Post-Condition**

* All zero elements in the input array **‘a’** have been moved to the end of the array while maintaining the order of the non-zero elements, which satisfies the post-condition of the program.

**Loop Variant**

* The expression **right<a.length** which ensures that the loop terminates when the right pointer reaches the end of the array.

**Loop Invariant**

* The code will check if the element at the current position of the right pointer ‘right’ is not zero using the condition **if (a[right] != 0)**. The first element is 0, so the condition is false and will skip to the next iteration of the loop.