Practise Tasks: Mar 31

JS

1. Given an array of distinct integers, replace every element with the least greater element on its right or with -1 if there are no greater elements.
2. Given an array of positive integers, find the smallest subarray's length whose sum of elements is greater than a given number k.

```
Input:  {1, 2, 3, 4, 5, 6, 7, 8}, k = 20
Output: The smallest subarray length is 3
Explanation: The smallest subarray with sum > 20
is {6, 7, 8}


Input:  {1, 2, 3, 4, 5, 6, 7, 8}, k = 7
Output: The smallest subarray length is 1
Explanation: The smallest subarray with sum > 7
is {8}


Input:  {1, 2, 3, 4, 5, 6, 7, 8}, k = 21
Output: The smallest subarray length is 4
Explanation: The smallest subarray with sum > 21
is {5, 6, 7, 8}


Input:  {1, 2, 3, 4, 5, 6, 7, 8}, k = 40
Output: No subarray exists
```

3. Given an integer array, rearrange it such that every second element becomes greater than its left and right elements. Assume no duplicate elements are present in the array.

```
Input:  {1, 2, 3, 4, 5, 6, 7}
Output: {1, 3, 2, 5, 4, 7, 6}

Input:  {9, 6, 8, 3, 7}
Output: {6, 9, 3, 8, 7}

Input:  {6, 9, 2, 5, 1, 4}
Output: {6, 9, 2, 5, 1, 4}
```

4. Given an integer array, find the largest subarray formed by consecutive integers. The subarray should contain all distinct values.

```
Input:  { 2, 0, 2, 1, 4, 3, 1, 0 }

Output: The largest subarray is { 0, 2, 1, 4, 3 }
```

5. Given a positive number n, find the next highest power of 2. If n itself is a power of 2, return n.

```
Input:   n = 20
Output: 32


Input:   n = 16
Output: 16
```

DSA

1. Given a binary tree, find the size of the largest BST (Binary Search Tree) in it.
2. Print all paths from the root to leaf nodes of a binary tree
3. Given a binary tree, write an efficient algorithm to check if it is height-balanced or not. In a height-balanced tree, the absolute difference between the height of the left and right subtree for every node is 0 or 1.
4. Given a binary tree, determine whether it is a BST.