1. Write a function that takes N and returns 1 if N is a member of Fibonacci sequence else returns 0.
2. Write a function that takes a number X and array of N elements and prints all pairs, which sum to X.
3. Write another function, which does the above but assumes array is sorted.
4. Write a function, which takes an array and number of elements as an argument and sorts the array using insertion sort.
   Function Prototype: **void insertionSort(int arr[], int N);**
5. Write a function for binary search. It should return the index of the element in the array or -1 if not found.
   Function Prototype: **int binarySearch(int arr[], int N, int X);**
6. Write a function, which takes an array and removes duplicates from the array while keeping the order of non-duplicate elements the same as the original array.
   Function Prototype: **void removeDuplicates(int arr[], int N);**
7. Convert find median of two sorted arrays into a function.
   Function Prototype: **int getMedianValue(int arr1[], int arr2[], int N, int M);**
8. You are given with an array of negative and positive numbers. Write an function to find the index at which the array should be divided into 2 sub-arrays in such a way that the difference between the sum of the 2 sub-array is maximum.
   Example –
   Input – [2, -4, 3, 1, -6, -1, 2, 7]
   Output – 5
   [**2, -4, 3, 1, -6, -1]** and **[2, 7]**. The difference is **9 – (-5) = 14**, which is maximum