Longest Bitonic Subsequence

Given an array of positive integers. Find the maximum length of Bitonic subsequence.
A subsequence of array is called Bitonic if it is first strictly increasing, then strictly decreasing.

Example 1:

```
Input: nums = [1, 2, 5, 3, 2]
Output: 5
Explanation: The sequence {1, 2, 5} is
increasing and the sequence {3, 2} is
decreasing so merging both we will get
length 5.
```

Example 2:

```
Input: nums = [1, 11, 2, 10, 4, 5, 2, 1]
Output: 6
Explanation: The bitonic sequence
{1, 2, 10, 4, 2, 1} has length 6.
```

Your Task:
You don't need to read or print anything. Your task is to complete the function **LongestBitonicSequence()** which takes the array nums[] as input parameter and returns the maximum length of bitonic subsequence.

Expected Time Complexity: $O(n^2)$
Expected Space Complexity: O(n)

Constraints:
$1 \leq$ length of array $\leq 10^3$
$1 \leq arr[i] \leq 10^6$