Pseudocode LongestIncreasingSubsequence

1. The code begins by defining a package called practiceProject2 and importing necessary classes (ArrayList, Arrays, and List) for the program.

2. A class named LongestIncreasingSubsequence is defined, which contains the main method and the findLongestIncreasingSubsequence method.

3. Inside the main method, an array of integers called numbers is created and initialized with some values.

4. The findLongestIncreasingSubsequence method is called with the numbers array as an argument, and the resulting longest increasing subsequence is stored in the lis list.

5. Finally, the longest increasing subsequence is printed to the console.

6. The findLongestIncreasingSubsequence method takes an array of integers numbers as input and returns a list containing the longest increasing subsequence of numbers.

7. It begins by initializing some variables: n to the length of numbers, and two arrays dp and prev, both of size n.

8. The prev array is filled with -1 values using the Arrays.fill method. This is done to indicate that initially, there is no previous element for any element in the subsequence.

9. The first element of the dp array, dp[0], is set to 1 since the longest increasing subsequence ending at the first element has a length of 1.

10. Two nested loops are used to iterate over the elements of numbers and calculate the length of the longest increasing subsequence ending at each index.

11. The outer loop runs from the second element (i = 1) to the last element of numbers. Inside this loop, the `dp[i]