import java.util.ArrayList;

import java.util.Arrays;

import java.util.List;

public class LongestIncreasingSubsequence {

public static void main(String[] args) {

// int[] numbers = {56, 32, 9, 47, 21, 70, 41, 60, 82};

int[] numbers1 = {78, 22, 98, 17, 51, 70, 41, 85, 65};

List<Integer> longestIncreasingSubsequence = findLongestIncreasingSubsequence(numbers1);

System.out.println("Longest Increasing Subsequence: " + longestIncreasingSubsequence);

}

public static List<Integer> findLongestIncreasingSubsequence(int[] numbers) {

int n = numbers.length;

int[] lengths = new int[n];

Arrays.fill(lengths, 1);

int[] previousIndices = new int[n];

Arrays.fill(previousIndices, -1);

int maxLength = 1;

int endIndex = 0;

for (int i = 1; i < n; i++) {

for (int j = 0; j < i; j++) {

if (numbers[i] > numbers[j] && lengths[i] < lengths[j] + 1) {

lengths[i] = lengths[j] + 1;

previousIndices[i] = j;

if (lengths[i] > maxLength) {

maxLength = lengths[i];

endIndex = i;

}

}

}

}

List<Integer> longestIncreasingSubsequence = new ArrayList<>();

while (endIndex >= 0) {

longestIncreasingSubsequence.add(0, numbers[endIndex]);

endIndex = previousIndices[endIndex];

}

return longestIncreasingSubsequence;

}

}