A subsequence of a sequence (for example, an array, linked list, or string), obtained by removing zero or more elements and keeping the rest in the same sequence order. A subsequence is called a substring if its elements are contiguous in the original sequence. For example:

- SUBSEQUENCE, USEQU and the empty string ε are all substrings of the string SUBSEQUENCE;
- SBSQNC, UEQUE, and EEE are all subsequences of SUBSEQUENCE but not substrings;
- QUEUE, SSS, and FOOBAR are not subsequences of SUBSEQUENCE.

Describe recursive backtracking algorithms for the following problems. Don’t worry about running times.

1. Given an array A[1..n] of integers, compute the length of a longest increasing subsequence. A sequence B[1..ℓ] is increasing if B[i] > B[i − 1] for every index i ≥ 2. For example, given the array

   \[\langle 3, 1, 4, 1, 5, 9, 2, 6, 5, 3, 5, 8, 9, 7, 9, 3, 2, 3, 8, 4, 6, 2, 7 \rangle\]

   your algorithm should return the integer 6, because \langle 1, 4, 5, 6, 8, 9 \rangle is a longest increasing subsequence (one of many).

2. Given an array A[1..n] of integers, compute the length of a longest decreasing subsequence. A sequence B[1..ℓ] is decreasing if B[i] < B[i − 1] for every index i ≥ 2. For example, given the array

   \[\langle 3, 1, 4, 1, 5, 9, 2, 6, 5, 3, 5, 8, 9, 7, 9, 3, 2, 3, 8, 4, 6, 2, 7 \rangle\]

   your algorithm should return the integer 5, because \langle 9, 6, 5, 4, 2 \rangle is a longest decreasing subsequence (one of many).

3. Given an array A[1..n] of integers, compute the length of a longest alternating subsequence. A sequence B[1..ℓ] is alternating if B[i] < B[i − 1] for every even index i ≥ 2, and B[i] > B[i − 1] for every odd index i ≥ 3. For example, given the array

   \[\langle 3, 1, 4, 1, 5, 9, 2, 6, 5, 3, 5, 8, 9, 7, 9, 3, 2, 3, 8, 4, 6, 2, 7 \rangle\]

   your algorithm should return the integer 17, because \langle 3, 1, 4, 1, 5, 2, 6, 5, 8, 7, 9, 3, 8, 4, 6, 2, 7 \rangle is a longest alternating subsequence (one of many).