DP Algorithm for Longest Increasing Substring

Given $X = x_1, x_2, \ldots, x_n$, to find the *length* of the longest increasing substring:

- $M[1] := 1$
- For $i = 2$ up to n do:
    If $x_{i-1} \geq x_i$ then $M[i] := 1$
    Else $M[i] := 1 + M[i-1]$
- Return $\max\{M[1], M[2], M[3], \ldots, M[n]\}$

Note: to find the actual longest increasing substring, you have to do backtracing.