Finding a Longest Increasing Substring

Example: $X = 1, 3, 1, 8, 2, 4, 9, 2, 10, 3$ (so $x_1 = 1, x_2 = 3$, etc.)

How can we design an algorithm to solve this problem?

Let $M[i]$ denote the length of the longest increasing substring that ends at x_i.

So:

- $M[1] = 1$
- $M[2] = 2$
- $M[3] = 1$
- $M[4] = 2$ (why isn't it 3?)

Class exercise:

1. calculate $M[i]$ for $i = 5, 6, 7, 8, 9, 10$.
2. What is the longest increasing substring for X?
3. What index does it end at?
4. What do you see for $M[i]$ for that index i?