به نام او

Quiz 2

Question:

Given an array A of n integers, we want to find the length of the longest increasing subsequence in $O(n^2)$ using dynamic programming (DP.)

Example: For [10, 9, 2, 5, 3, 7, 101, 18], the longest increasing subsequence is [2, 3, 7, 101], so the length is 4.

You should formulate a recursive equation to approach the problem and then provide pseudocode which uses DP to solve it.

Answer:

Recursive equation (60 pts) and pseudocode (40pts).

Solution 1:

Suppose $DP[i]$ denotes the length of the LIS if we consider the i^{th} element is part of the solution. Clearly, $DP[0] = 1$. The recursive equation for $DP[i]$ would be:

$$DP[i] = \max\left\{1,\ \max_{\substack{j<i\\ A[j]<A[i]}} \{DP[j]\} + 1\right\}$$

The solution to the problem would be $\max_{0\leq i<n}\{DP(i)\}$. The pseudocode for this problem is listed below:

```
void calculate_LIS(A) {
    DP[0] = 1
    n = length(A)
    for i = 1 to n - 1 {
        DP[i] = 1
        for j = 0 to i - 1 {
            if A[j] < A[i] {
                DP[i] = max(DP[i], DP[j] + 1)
            }
        }
    }
    return max(DP[i], 0<=i<n)
}
```

Solution 2: Alternatively, you could sort the given array and compute the LCS between the sorted and given array.