Name:

Advanced Programming, II Semester, 2014–2015

Quiz 7, 15 April 2015

Answer all questions in the space provided. Use the reverse for rough work, if any.

At the end of its fifth successful season, the Siruseri Premier League is planning to give an award to the Most Improved Batsman over the five years. For this, an Improvement Index will be computed for each batsman. This is defined as the longest subsequence of strictly increasing scores by the batsman among all his scores over the five seasons. For example, if the scores for a batsman over the five seasons are $[20, 23, 6, 34, 22, 52, 42, 67, 89, 5, 100]$, his improvement index is 7 based on the subsequence $[20, 23, 34, 52, 67, 89, 100]$.

1. let $S[1..n]$ denote a sequence of n scores for which the improvement index is to be calculated. For $1 \leq j \leq n$, let $I(j)$ denote the improvement index for the prefix of scores $S[1..j]$ ending at $S[j]$.

   Which of the following is a correct recursive formulation of $I(j)$?

   (a) $I(1) = 1$
   For $j \in 2, 3, \ldots, n$, $I(j) = 1 + \max\{I(k) \mid 1 \leq k < j, S[k] > S[j]\}$

   (b) $I(1) = 1$
   For $j \in 2, 3, \ldots, n$, $I(j) = 1 + \max\{I(k) \mid 1 \leq k < j, S[k] < S[j]\}$

   (c) $I(1) = 1$
   For $j \in 2, 3, \ldots, n$, $I(j) = \begin{cases} 1 + S[j-1], & \text{if } S[j-1] < S[j] \\ 1, & \text{otherwise} \end{cases}$

   (d) $I(1) = 1$
   For $j \in 2, 3, \ldots, n$, $I(j) = \begin{cases} 1 + S[j-1], & \text{if } S[j-1] > S[j] \\ 1, & \text{otherwise} \end{cases}$

   Answer: (b) Look for the longest sequence that can be extended by $S[j]$.

   (4 marks)

2. How would we evaluate this recursive definition using dynamic programming?

   (a) A one dimensional table t of size n, to be filled from $t[1]$ to $t[n]$

   (b) A one dimensional table t of size n, to be filled from $t[n]$ to $t[1]$

   (c) A two dimensional table t of size $n \times n$, to be filled row-wise from $t[1][1]$ to $t[n][n]$.

   (d) A two dimensional table t of size $n \times n$, to be filled row-wise from $t[n][n]$ to $t[1][1]$.

   Answer: (a) The recursive function has a single argument, so we need only a one dimensional table. The base case is $I(1)$, so start with $T[1]$ and work up to $T[n]$.

   (3 marks)

3. How much time will it take to evaluate this recursive definition using dynamic programming?

   (a) $O(n)$
   (b) $O(n \log n)$
   (c) $O(n^2)$
   (d) $O(n^3)$

   Answer: (c) Each entry $I(j)$ requires scanning $I(1)$ to $I(j-1)$, so the time taken is $1 + 2 + \cdots + n - 1$ which is $O(n^2)$.

   (3 marks)