Imperial College London – Department of Computing

MSc in Computing Science

580: Algorithms
Tutorial: Dynamic Programming

1. The array $A = [A_1, \ldots, A_N]$ contains N integers.

   (a) A *prefix subarray* of A is any continuous subarray that starts with $A[1]$. Write a $\Theta(N)$-time algorithm to find the greatest sum of any prefix subarray of A.

   (b) The greatest sum of *any* continuous subarray of A can be found as follows. For each position i in the array, find s_i, the maximum sum of any continuous subarray starting at i. The solution is then the maximum of those s_i values. There are N such values, so this method will take $\Theta(N^2)$ time using your answer for (1a). This is referred to as a *naive* solution.

   By considering the structure of the naive solution, design a $\Theta(N)$-time solution to the problem.