CIS 621, Algorithms and Complexity
Winter 2007

Assignment 3

due 2007 February 8, Thursday

1. Recall the Knapsack problem, where we have a knapsack of capacity \( W \) and \( n \) items with weights \( w_1, w_2, \ldots, w_n \) and values \( v_1, v_2, \ldots, v_n \). In class we saw an \( O(nW) \) dynamic programming algorithm solving the version of this problem that allows repetition. You are to solve the version of the problem with no repetition.

   optional: this problem looks to be a natural one on which to try memoization.

2. exercise 15-1, p 364.

3. You are given a sequence \( s \) of \( n \) characters, which may be a piece of text from which all punctuation has been removed (for example, \( s \) may be “visionsofsugarplumsdancing...” or it may be simply garbage). You wish to reconstruct the sequence from a dictionary - that is, you are given a function \( \text{dict}(w) \) which returns true iff \( w \) is a valid English word. You may assume that \( \text{dict} \) runs in constant time. Give a dynamic programming algorithm which will determine whether the sequence \( s \) can be decomposed into a series of valid words.

4. optional We did exercise 15-6, p 368, in class, but perhaps not very efficiently. Try to do it more efficiently.

Notes:

- Q1 is exercise 16.2-2, p 384.
- For Q3, optionally make your algorithm give the series of valid words.