Dynamic Programming practice problems:

For each, develop a recursive solution, convert it to a DP solution, then write the traceback routine. For each show for four smaller distinct problems that the recursive solution returns the same answer as the DP, and show the traceback routine working for four diverse of problems.

1) Given a dictionary of keywords D and a string A, of length n, determine if there exists some subset of keywords that when concatenated together make up A. Key words maybe used multiple times.

2) You are tasked with developing a game played on a 3 dimensional board of cubes sized n X m X p (n, m and p are integers). Players have characters that occupy a cube and can move to other cubes with moves that change the coordinates of the character. These moves are described as a list of 3D tuples called M. Each tuple is (change in x, change in y, change in z). Moves are not allowed that take the character outside of the board. Solve the following problem:
    a. Given n,m,p and M, and a starting location (a, b, c), determine all locations that can be reached from the starting location

3) Given two knapsacks of size K1 and K2, and a set of objects each of size[i], determine (K1 and K2 and the contents of s[i] are all integers) if there exists a subset of the objects that fit exactly into K1 and K2. You can use each object any number of times.

4) Given a game board sized n by m squares, were each square contains V[I,j] dollars. A player may start at any location in the first column (when n = 0) and then move either diagonally up or diagonally down from the current square so long as the player stays within the board. When the player is in a square I,j they collect the money at that square V[I,j]. Determine the maximum amount of money that can be collected.

5) Given a sequence of n coins 0 … n-1, each with value V[i], 0<=i<n, assume n is even. You are playing a game against an opponent as follows. At your turn, you pick a coin from either side of the sequence and take the value of that coin. Then your opponent gets to pick from each side and take that value. Both players are trying to maximize their value. What is the maximum value that may be obtained by the player that moves first?