Problem Set 5: Dynamic Programming
Due: Dec 15, 2020, 8pm

Note: For all problems, if you include pseudocode in your solution, please also include a brief description of what the pseudocode does.

1. **[Shortest Travel Sequence] (10 points)**

   Two friends f_1 and f_2 plan to travel together. Each one has his/her preferred travel route. In this problem, we only consider the sequence of the tourist attractions. For example, f_1 wants to have a trip to visit BCAEF following the very accurate sequence. In addition, they do not care the tourist attractions inserted in their preferred sequences. Please design an algorithm to output the length of the shortest valid travel sequence for the two friends with given two travel sequences.

   For example, the sequence of f_1 is ABCB and that of f_2 is DBC. Your algorithm should output 5, as ADBCB or DABCB are two valid shortest travel sequences and all are in length of 5.

   [We are expecting: pseudocode, time complexity analysis and an informal justification that it is correct.]

2. **[Big Bang] (10 points)**

   One interesting function introduced in Smartisan OS (a flavor android system) is *Big Bang*. It can break the text in the screen into several small pieces. Let's design an algorithm to implement this function. Assume that you have a dictionary of words, the task is to break a sequence of characters into pieces of words contained in the dictionary. For example, the dictionary is {alg, algorithm, is, interest, interesting, interested}. For the input of "algorithmisinteresting", your algorithm should output the smallest set of words: "algorithm", "is", "interesting". If the sequence cannot be fully divided (e.g., "algithmisinteresting"), just output "Incomplete Dict!".

   [We are expecting: pseudocode, time complexity analysis and an informal justification that it is correct.]

3. **[Shortest Path in Grid Space.] (10 points)** Let us consider the shortest path in the grid space. Given a $n \times n$ cost table G, each $G[i, j]$ indicates the positive travel cost in the corresponding grid cell $C[i, j]$ in i-th line and j-th column. Please design an algorithm to output the shortest path from the left-top corner, $C[0, 0]$, to the right-bottom corner, $C[n-1, n-1]$.

   [We are expecting: pseudocode, time complexity analysis and an informal justification that it is correct.]

4. **[Fish fish eat eat fish.] (10 points)** Plucky the Pedantic Penguin enjoys fish, and he has discovered that on some days the fish supply is better in some lake and some days the fish supply is better in other lake.

   He has access to a 2-D table S, where $S[i, j]$ is the number of fish he can catch in Lake j on day i. Assume that $S[i, j]$ is positive integer for any $i = 1, \ldots, n$ and $j = 1, \ldots, m$.

   Plucky's goal is to have as many fish as possible at the end of day n. If he happens to be at Lake x on day i and wants to be at Lake y on day $i + 1$, he may pay $C(x, y)$ fish to a polar bear who can take him from Lake x to Lake y overnight; the same is true if he wants to go from Lake y back to Lake x. Here $C(x, y) > 0$ is a positive integer for any $x, y \in [1, m]$ and $x \neq y$. Assume that before day 1 begins, Plucky is at Lake x, and he has zero fish.

   Plucky will save all the fish he gets until the end of day n (at which point he will feast), but the polar bear does not accept credit. So Plucky must have the fish on hand in order to pay the polar bear; he must pay *before* he travels.

   In this question, you will design a dynamic programming algorithm that finds the maximum number of fish that Plucky will have on day n. Do this by answering the two parts below. [hints: You can first try to solve a problem of only two lakes.]

(a) (5 points) What sub-problems will you use in your dynamic programming algorithm? What is the recursive relationship which is satisfied between a problem and the sub-problems? What is the base case for this recursive relationship? Justify your answer.

[We are expecting: a formal definition of your sub-problems, as well as a recursive relationship that they satisfy. We are also expecting an informal justification that your recursive relationship is correct given your definitions.]

(b) (5 points) Write pseudocode for a dynamic programming algorithm that takes as input of fish supply table S, travel cost table C, m, and n, and returns the maximum number of fish that Plucky can eat on day n.

[We are expecting: the pseudocode, a brief explanation (one or two sentences) about why it works, and explain (also one or two sentences) that its running time.]

5. **[There is no problem 5.] (0 points)** If you finish all of the above and still want more practice with dynamic programming, try the problems in Chapter 15 of CLRS or go to leetcode to do more practice.