Problem Set 5: Design and Analysis of Algorithms

When a dynamic programming algorithm is presented, follow the given procedure:

- Define sub-problem(s) and explain how they help in solving the given problem.
- Design a recurrence relation for the sub-problem and prove its correctness.
- Give pseudocode to show how the sub-problems are computed.

Problems

1. Assume you are in charge of the Security management team. You are supposed to pick a project for your team every week. You have the option to pick a low-risk project or a high-risk project or no project at all. For week i, your team will earn h_i points if they do the high-risk project and l_i points if they do the low-risk project. (Your team gets zero points if they are not doing any project). Also, if you select a high-risk project in week i, the team is required not to have done any project in week $i-1$.

   Assume the values for h_i and l_i are known for $1 \leq i \leq n$. Give an algorithm to find the maximum total points you can earn in n weeks.

2. Let $IsWord(S)$ be a function that takes a string S as input and outputs TRUE if and only if S is a valid English word. Assume $IsWord(S)$ takes constant time to run. Given a string S, design an algorithm to decide if S can be broken into one or more valid English words. For example, if S is "ARTISTOIL", the algorithm should return TRUE since S can be broken down to "ARTIST" and "OIL".

3. In the above problem, modify the algorithm so that it returns the maximum number of ways in which a string can be broken down into valid English words. Therefore, if S is "ARTISTOIL", the algorithm should return 2 since S can be broken down into "ARTIST" and "OIL" and "ART", "IS", and "TOIL".

4. In the above problem, modify the algorithm so that it returns the minimum k such that the string can be broken down into k valid English words. Therefore, if S is "ARTISTOIL", the algorithm should return 2 since S can be broken down into "ARTIST" and "OIL".

5. Given two strings A, B of length n, decide whether A and B can be partitioned into words at the same indices. For example, the strings "HEARTANDBODY" and "STARTONEWEEK" can be partitioned into words at the same indices as follows:

   HEART AND BODY
   START ONE WEEK

   Assume that whether a given string is a valid word can be checked in constant time.

6. Given a string S, give an algorithm to find the minimum value of k such that S can be partitioned into k substrings that are palindromes. For example, the string "BANANAAXA" can be partitioned into "B.ANANA.AXA".

7. In a previous role, you worked as a cashier in the forgotten Antarctic colony of Nadiria, where much of your day involved providing customers with change. Due to the rarity and high value of paper in Antarctica, cashiers were legally mandated to minimize the number of bills used when giving change. Owing to the numerological preferences of one of its founders, the currency of Nadiria, known as Dream-Dollars, was issued in the following denominations: \$1, \$4, \$7, \$13, \$28, \$52, \$91, and \$365.

   (a) The greedy algorithm for giving change selects the largest available bill that does not exceed the remaining amount. For instance, to make \$122 using the greedy algorithm, one would first take a \$91 bill, followed by a \$28 bill, and finally three \$1 bills. Provide an example where the greedy algorithm results in a higher number of bills than the optimal solution.

   (b) Develop and analyze a recursive algorithm that determines the minimum number of bills needed to make an integer amount k in Dream-Dollars. Ensure correctness without focusing on efficiency.

   (c) Propose a dynamic programming solution to compute the minimum number of bills required to form an amount k in Dream-Dollars efficiently.