CS 180 Homework 4

Due not later than *Sunday, August* **8, 2021**

Problem. On the planet Sigma, robots excavate chunks of a very precious cristaline. These chunks can be divided into smaller part only on the Earth. Once a month, a spaceship comes to Sigma to bring cristaline to the Earth. The spaceship cannot take more than k pounds of the load. Naturally, it is desirable to take as much cristaline as the spaceship can.

  (50 pts) Write an algorithm that will allow bringing the maximal amount of cristaline to the Earth. Prove that it is correct and estimate its time complexity.

  (100 pts) Write an efficient algorithm that will allow bringing the maximal amount of cristaline to the Earth. Prove that it is correct and estimate its time complexity.

Efficient algorithm:

Subset-Sum(n, K)
    k = maximum pounds allowed of the load
    Array M[0…n, 0 . . . K]
    Initialize M[0,k] = 0 for each k = 0, 1, … , K
    For i from 1 to n do:
        For k from 0 to K do:
            If k < k_i: **//k_i denotes the size of i-th cristaline**
                m[i,k] = m[i-1, k] //the new cristaline is more than the weight limit
            Else:
                m[i,k] = max(m[i-1,k], k_i + m[i-1,k-k_i])
      Return M[n, K]

Proof of Correctness:
We declare that $a[i,0] = a[0,k] = 0$ for all i, k
When computing $a[i', k']$, by induction hypothesis, we have $a[i' - 1, k']$, $a[i' - 1, k' - k_{i'}]$ are already computed correctly. Then algorithm considers the optimal value for item i' in subset as $a[i' - 1, k' - k_{i'}]+k_i$ and for item i' not in spaceship as $a[i' - 1, k']$. Therefore, the value at $a[i', k']$ is correct.

Basically, we select any subset of the items such that there is no other item that can be feasibly chosen. And, we also select the largest item. Out of those 2, choose the subject which yields the larger total pounds.

Time Complexity: O(n*k); where n is the total number of cristalines in the input and k is a constant of pounds limit (capacity), then O(n)