Istanbul Medipol University
Department of Computer Engineering
Medipol COE223389 Algorithm Analysis
Spring 2021, ASSIGNMENT #4
Due "28 May 2021 @ 11:59PM (23:59)"

Objective: To get familiar with and practice dynamic programming and greedy algorithms.

You must solve all the questions. Submit with your solutions a cover page where you are expected to specify for each question the names of students with whom you discussed. In addition to the complexity analysis parts, submit your programs, your test data and your results.

Question 1 (50 points): Dynamic Programming

Implement the algorithm for an optimal parenthesization of a matrix chain product as discussed in the class.

Use the following recursive function as part of your program to print the outcome, assume the matrixes are named $A_1, A_2, ..., A_n$.

```
PRINT-OPTIMAL-PARENS(s, i, j) {
if (i=j) then
        print "A"i else {
        print "("
        PRINT-OPTIMAL-PARENS(s,i,s[i, j])
        PRINT-OPTIMAL-PARENS(s, s[i, j] + 1, j)
        print ")"
} }
```

a- Test your algorithm for the following cases:

1. Find and print an optimal parenthesization of a matrix-chain product whose sequence of dimensions is $< 5, 10, 3, X, 12, 5, 50, Y, 6 >$.
2. Find and print an optimal parenthesization of a matrix-chain product whose sequence of dimensions is $< 5, 10, 50, 6, X, 15, 40, 18, Y, 30, 15, Z, 3, 12, 5 >$.

3. Find and print an optimal parenthesization of a matrix-chain product whose sequence of dimensions is $< 50, 6, X, 15, 40, 18, Y, 5, 10, 3, 12, 5, Z, 40, 10, 30, 5 >$.

Where each of X, Y and Z is a two digits number to be extracted from the leftmost 6 digits of your TC ID (from left to right) by the same way you did in the midterm.

b- Find the complexity of your program

c- Show that the parenthesization algorithm is loop invariant.

Question 2 (50 points): Greedy Algorithms

Given an integer $V > 1$ and a list of m distinct integers $L[1..m]$ where $L[i] < V$, for $i = 1\ to\ m$ and $L[m] = 1$

Implement the greedy algorithm to represent V as the sum of products using minimum possible number of integers from L as discussed in class.

You may use another array $R[1..m]$ to keep track of the intermediate results. Then print the outcome using values from R and L. For instance, given $V = 69$, $m = 5$ and $L = [14, 8, 6, 5, 1]$. Then the algorithm will find $R = [4, 1, 0, 1, 0]$, which will lead to $V = 14 \times 4 + 8 \times 1 + 6 \times 0 + 5 \times 1 + 1 \times 0$, that is, $V = 14 \times 4 + 8 \times 1 + 5 \times 1$

a- Test your algorithm for three cases of your choice. In each of the three cases V should be greater than 150, m should be at least 6, and the maximum value in L should be in the range $[10, 20]$. Keep in mind that the integers in L must be distinct.

Two of your cases should produce minimum combinations, while the third example should not produce the actual minimum combination. In other words, the third example will illustrate that the algorithm does not work for every given case.

b- Find the complexity of your program

c- Show that the implemented greedy algorithm is loop invariant.