Homework Assignment No. 3

CSCI 4470/6470 Algorithms, CS@UGA, Spring 2018

Due Thursday March 1, 2018

The answers must be word-processed or typed. You may substitute formulae and figures with hand-writings. Your submitted algorithms should be in the pseudo-code, not in any specific programming language. Answers deviating from these requirements will be returned without grading.

The answers must be the student’s own work. Idea sharing and referencing to others’ work (including those online) are not allowed. Plagiarism and other forms of academic dishonesty will be handled within the guidelines of the Student Handbook and reported to the University.

Four questions with total 110 points.

1. (30 points) Consider the Casino Dice Decoding Problem (ref. pages 35-38 of lecture note 3, NewNote3.pdf). Note that the problem resembles the Assembly line problem. Fill out a dynamic programming table that computes functions \( p(S, i, F) \) and \( p(S, i, L) \) for sequence of digits \( S = 6661234 \). You can choose to write a program to or manually calculate for the table. Show how the computation can tell the most probable hidden sequence of dice that may have been used to roll the given 7 digits.

2. (20 points) Fill out a dynamic programming table that computes for the 0-1 knapsack problem on the following input data: \( B = 10, i = 1, 2, 3, 4 \), and

\[
\begin{align*}
  s_1 &= 3, v_1 = 50 \\
  s_2 &= 5, v_2 = 40 \\
  s_3 &= 3, v_3 = 40 \\
  s_4 &= 2, v_4 = 10
\end{align*}
\]

3. (40 points) Consider following Article Grading problem. A teacher grades students’ articles based on number of letter corrections on an article. The teacher may correct an
article with the following four types of editing:

\[
\begin{align*}
\text{a letter is kept} & \quad +5 \text{ points} \\
\text{a letter substitution} & \quad -2 \text{ points} \\
\text{a letter insertion} & \quad -3 \text{ points} \\
\text{a letter deletion} & \quad -3 \text{ points}
\end{align*}
\]

You are required to formulate a dynamic programming solution that can compute the maximum number of points for any article \(X\) based on comparison between \(X\) and the edited version \(Y\). This problem can be thought as a generalization of the LCS problem, where the situation of "a letter is kept" is rewarded 1 point while no penalty applies on a letter "substitution", "insertion", or "deletion".

Let the original article and edited version be \(X\) and \(Y\), respectively. Follow the idea used in the problem analysis for LCS, consider any prefix \(x_1 \ldots x_i\) of \(X\) and any prefix \(y_1 \ldots y_j\) of \(Y\). Assume \(Grade(i,j)\) to be the maximum score gained when prefix \(x_1 \ldots x_i\) has been edited to \(y_1 \ldots y_j\). There are four possibilities of editing from letter \(x_i\) to letter \(y_j\):

1. \(x_i\) is kept (only if \(x_i = y_j\));
2. \(x_i\) was substituted with \(y_j\);
3. \(y_j\) was inserted after the position \(x_i\);
4. \(x_i\) was deleted (\(x_i\) does show in the edited version anymore);

**Part A.** Based on these 4 cases, you are required to formulate \(Grade(i,j)\) into a recursive function, i.e., being defined by the same function \(Grade\) (with smaller parameter values, however).

**Part B.** Write an iterative algorithm to compute for function \(Grade\); the algorithm needs to include steps to remember the choices of editing.

4. **(20 points)** Consider a coin change problem with the coin set \(\{C_1, C_2, C_5\}\), where \(C_k\) represents the coin of \(k\) cents. Prove that this problem has the greedy choice property. You may have to prove for the cases of cents \(\geq 5\), of cents \(2 \leq \text{ but } < 5\), and of cents \(\leq 1\), respectively.