CIS 455 – Homework #2

Due on myCourses by 11:59pm on Monday, Oct. 5, 2015

Instructions: There are 7 problems worth a total of 100 points.

Submit on myCourses: solutions to the written parts, any source code and screenshots of sample runs which thoroughly verify the correctness of your code.

Document and indent your programs properly. You will be graded on both your solutions and your ability to show their correctness.

If you feel it would help, you are encouraged to work together on homework. But remember that you must submit your own work, as the point of the homework is to learn the material. *If you do work with others on homework, you must write the names of those you worked with on your homework.*

Late Homework will be penalized!

Problem 2-1. (20 points) Jones & Pevzner, Problem 4.1, page 119.

1. Write an algorithm in pseudocode that given a set X, calculates the multiset ΔX.
2. Implement the above algorithm in Matlab, provide your source code and testing results to submit. Matlab is available in Dion 303 and on the Virtual Computing Lab: http://www.umassd.edu/cits/vcl

Problem 2-2. (20 points) Jones & Pevzner, Problem 4.2, page 119.

Consider partial digest $L = \{1,1,1,2,2,3,3,3,3,4,4,5,5,5,6,6,6,9,9,10,11,12,15\}$.

Solve the Partial Digest problem for L (i.e., find **all** X such that ΔX = L).

*Make sure you find **every** X* such that ΔX = L.

Hint: You can solve this problem by doing **either one** of the following:

(1) Manually follow the steps of PARTIALDIGEST(L) on page 90.

or

(2) Write a program that implements ANOTHERBRUTEFORCEPDP on page 88 and apply it to L.

Problem 2-3. (10 points) Jones & Pevzner, Problem 4.5, page 119.

Prove that the sets $U \oplus V = \{u + v : u \in U, v \in V\}$ and $U \Theta V = \{u - v : u \in U, v \in V\}$ are homometric for any two sets U and V.

Problem 2-4. (18 points) Jones & Pevzner, Problem 4.9, page 121 (*read the DDP description on pages 119-121*).

1. Come up with a brute force algorithm for the DDP and show pseudocode of your algorithm.
2. Suggest a branch-and-bound approach to improve its performance.

Problem 2-5. (12 points) Jones & Pevzner, Problem 4.12, page 121.
Given a long *text* string *T* and one shorter *pattern* string *s*, find the first occurrence of *s* in *T* (if any).

1. Show pseudocode of your algorithm.
2. What is the complexity of your algorithm?

Problem 2-6. (10 points) Jones & Pevzner, Problem 4.16, page 122.

Problem 2-7. (10 points) Jones & Pevzner, Problem 4.17, page 123.