**EE 367 Spring 2018**

**Homework 5 Total Points = 16**

**Problem 5.1-2** (page 117, 1 pt)

**Problem 5.2-2** (page 122, 1 pt)

**Problem 5.2-4** (page 122, 1 pt)

**Problem 5.2-5** (page 122, 1 pt)

**Problem 5.3-2** (page 128, 1 pt) *[ Hint: Consider the case when n = 3. ]*

**Problem 5.3-3** (page 128, 1 pt)

**Problem 5.3-5** (page 129, 1 pt)

**Problem 5.3-6** (page 129, 1 pt)

**Problem 5-2** (page 143 parts a, b, c, and d (skip parts e-f, 2 pts, 0.5 pts per part)

**Problem 13.1-3** (page 311, 1 pt)

**Problem 13.2-4** (page 314, 1 pt) *[Hint: the right going train are the string of nodes that start at the root and keeps going down the right child]*

**Problem 13.3-2** (page 322, 1 pt) When drawing the red-black trees, draw the following for red and black nodes:

41

(red)

41

(blk)

**Problem 13.3-5** (page 322, 1 pt)

**Problem 13.4-3** (page 330, 1 pt) Draw the red and black nodes as in Problem 13.3-2.

**Problem A** (1 pt).

Attached is a program strlength.c that reads in string of lower case alphabets, and finds the longest substring of a single character. For example,

aaabaabbababbbba

has the longest substring bbbb which starts at position 12 and ends at position 15. The algorithm is implemented in the function strlength().

Modify the function strlength(), so that it finds the longest substring with at most two distinct characters. For example,

aaabcbbcbacca

has the longest substring bcbbcb which starts at position 4 and ends at position 9.

Your algorithm should have O(n) time complexity, and you should give an explanation of why this is so.

Upload your program in laulima.

The program reads a data file, which contains the string. The format of the data file is

<Length of string>

<String of lower-case characters>

Example:

13

aaabcbbcbacca