CS 323.000: Data Structures and Algorithms — Handout #7
Emory University, Math/CS Dept. — April 23, 2012

Homework #5: Strings and Flows
Due Date: written 5pm Monday 4/30, program 9pm Thursday 4/26

The program part (BWT.java) was introduced in class, see hw5/Notes.txt.

These written problems are largely review for our final exam (except for the last, they are all "work out this algorithm" problems). You only need to write up **three** of these on paper, and turn them in by 5pm Monday. I'll present solutions at our review session Wednesday (5/2, reading day). I'll also post the solutions online, for those who might miss the review session. Our final exam is 4:30-7pm Friday 5/4.

Problem 1. This is based on exercises from page 726.

1(a). Give a trace (like bottom of page 707) of LSD sort on these strings:

```
no is th ti fo al go pe to co to th ai of th pa
```

1(b). Give a trace (like page 714) of MSD sort on these strings.

```
now is the time for all good people to come to the aid of their party
```

1(c). Indicate which characters of the above strings were NOT examined by MSD sort.

Problem 2. Consider these strings again:

```
now is the time for all good people to come to the aid of their party
```

2(a). Draw the R-way trie that results when these strings are inserted into an empty trie. (Omit null links. Insertion order does not matter here. Ignore duplicates.)

2(b). Draw the TST that results when these strings are inserted into an empty TST. (Insertion order does matter here. Ignore duplicates. No rebalancing.)

Problem 3. Consider the pattern string P = "`BABBABAA`" (without the quotes).

3(a). Draw the KMP DFA for P.

3(b). Draw the KMP NFA for P. (The NFA construction is not well described in your textbook. See the files in `0416/`, including an example drawing on page 29 of the slides.)

Problem 4. Consider the string S = "`YABBADABBADOO!`" (without the quotes).

4(a). Work out the Huffman code for S. Write down the code as a codeword table, and also as a binary trie. (Different students may get different answers here, since you may break ties in different ways.)

4(b). Write down the bitstring encoding of string S (just the string, not its length or the trie).

4(c). Using the book representation (with 8 bits per symbol), how long is the bistring representing the code trie?

Problem 5. Consider the string S = "`ABABABAABAAB`" (without the quotes). In our initial LZW code table, suppose `'A'` is 41, `'B'` is 42, end-of-file is 80, and the first available code is 81.

5(a). Suppose we compress S. Show the complete sequence of code values emitted to compress S. Also, list the new entries in our code table (the first entry: `'AB'` is 81).

5(b). Suppose we decompress (expand) the sequence of code values found in the first part. Indicate when we make each new entry in our code table, and indicate the step(s) requiring the "tricky case".

Problem 6. Suppose we have an undirected graph G, with two distinguished vertices s and t.

6(a). Suppose each edge e of G has a weight $w(e)$. Describe how to compute a min-weight set of edges, whose deletion separates s from t. (Hint: this is an "easy" reduction to max-flow, mainly you convert the undirected graph G to a directed flow network.)

6(b). Suppose each vertex v of G has a weight $w(v)$. Describe how to compute a min-weight set of vertices, whose deletion separates s from t. (Hint: this is a harder reduction to max-flow. Represent each vertex v of G by two vertices v_0 and v_1 in your flow network, with an edge from v_0 to v_1 of capacity $w(v)$.)