Purdue University | Hambrusch
CS 580 | Spring 2010

Assignment 5

Due: Tuesday, March 9, 2010 (before class)
Midterm Exam: Wednesday, March 10, 8-10pm, Lawson B155 (no class on March 11)

1) (20 pts.)
(i) You are given two stacks A and B. Consider the following operations:

- Push(A,x): push element x into stack A
- Push(B,x): push element x into stack B
- multi-pop(A,k): repeatedly pop elements from A until either k elements have been popped or A is empty
- multi-pop(B,k): repeatedly pop elements from B until either k elements have been popped or B is empty
- Transfer(k): repeatedly pop elements from A and push them onto B until either k elements have been popped from A or A is empty

Assume you start off with A and B empty. Give a potential function Φ such that the amortized cost of every operation is O(1). Show that a potential is never less than the initial potential (which should be 0).

(ii) Consider a tree-based Union/Find implementation using the following rule for unioning two sets S_1 and S_2: each set maintains a quantity *degree* which represents the number of children of the root. When unioning two sets, the tree with the smaller degree is linked into the tree with the larger degree. The degree of the resulting new set is updated. Does this unioning rule ensure that every set is represented by a tree of height $O(\log n)$, where n is the number of elements in the set? Either give a proof or a counterexample (the counterexample should be in terms of n).

(iii) Let $G = (V, E)$ be a connected, undirected graph with weights on the edges. Let $T = (V, E_T)$ be a minimum-cost spanning tree of G.

- Suppose all the weights of the edges are incremented by a constant number c. Are the edges in E_T still the edges of the minimum-cost spanning tree? Either give a counterexample showing it is false or give an argument showing why it is true.
- Suppose every edge weight $w(u, v)$ is changed to $w(u, v)^2$. Are the edges in E_T still the edges of the minimum-cost spanning tree? Either give a counterexample showing it is false or give an argument showing why it is true.

2) (15 pts.) A palindrome is a string that reads the same forwards and backwards, like x, pop, noon, redivider, dammitimmad, madamimadam, or amanaplanacatahamayakayamahatacanalpanama. Any string can be broken into sequence of palindromes. The string bubbaseesabanana can be broken into palindromes in different ways, including:

bub + baseesab + anana
bub + b + a + sees + a + b + anana
b + u + bb + a + sees + aba + nan + a
b + u + bb + a + sees + a + b + anana
b + u + b + b + a + s + e + e + s + a + b + a + n + a + n + a

Describe and analyze an efficient algorithm to find the smallest number of palindromes that make up a given input string of length n. For example, given the input string bubbaseesabanana, the algorithm would return the integer 3. (Hint: use dynamic programming)

3) (15 pts.) You are given an $n \times n$ grid M. Certain grid locations contain treasures. In particular, for $M[i, j] = k$, location $[i, j]$ contains k treasures. A robot starts in the upper-left corner of the grid (i.e., $M[1, 1]$) and moves to the bottom left-hand corner. The robot can only move in two directions: right and down. If the robot visits a grid location with k treasures, the k treasures are picked up (there is no limit on how many treasures the robot can gather).

Describe an efficient algorithm to determine a path for which the number of treasures collected is a maximum. Output the path by listing its corners.