CS 473: Algorithms, Fall 2009
HBS 14

This HBS contains review problems for the final exam. Try to solve 3 of the 6 problems during the hbs. *Note that the problems are ordered by topic, not difficulty.*

Problem 1. An assignment for a computer graphics course requires students to write a program that rotates pictures by 90 degrees. Assume that the picture is square and has size $2^n \times 2^n$ pixels. Give a divide-and-conquer algorithm that makes 4 recursive calls, and 5 *block transfers* per call. A block transfer is a library routine that copies a square block of pixels from one location to another; this can be implemented fairly quickly. How many block transfers does your algorithm perform in total?

Problem 2 ([**3.10**]). Let $G = (V, E)$ be an unweighted, undirected graph and let u and v be two vertices of G. Describe a linear time algorithm to find the number of shortest paths from u to v. Note that we only want the number of paths as there may be an exponential number of them. Give an example graph with an exponential number of (u, v)-paths.

Problem 3 ([**6.6**]). You are tasked with a writing a "pretty print" algorithm for a word processor. You are given a sequence of words, $W = \{w_1, \ldots, w_n\}$ where word w_i consists of c_i characters. You are also given a maximum line length L. A *formatting* of W consists of a partition of the words in W into *lines*. In the words assigned to a single line, there should be a space after each word except the last; and so if $w_j, w_{j+1}, \ldots, w_k$ are assigned to one line, then we should have

$$\left[\sum_{i=j}^{k-1}(c_i + 1)\right] + c_k \leq L.$$

We will call an assignment of words to a line *valid* if it satisfies this inequality. The difference between the left-hand side and the right-hand side will be called the *slack* of the line—that is, the number of spaces left at the right margin.

Give an efficient algorithm to find a partition of a set of words W into valid lines, so that the sum of the *squares* of the slacks of all lines (including the last line) is minimized.

Problem 4. You are given a flow network G with source s and sink t. We call $(S, V \setminus S)$ an (s,t)-cut if $s \in S$ and $t \in T$. Define the intersection of two cuts $(S, V \setminus S)$ and $(S', V \setminus S')$ to be $(S \cap S', V \setminus (S \cap S'))$. Union is defined similarly. Show that the intersection of two minimum cuts is a minimum cut and that the union of two minimum cuts is a minimum cut.

Problem 5. Answer both questions on NP-completeness below.

- ([**8.17**]) You are given a directed graph $G = (V, E)$ with weights w_e on each edge $e \in E$. The weights can be negative or positive. The *Zero-Weight-Cyle Problem* is to decide if there is a simple cycle in G such that the sum of the edge weights on this cycle is exactly 0. Prove that this problem is NP-Complete.

- A SAT formula ϕ is said to be *monotone* if each clause has only positive literals. For example,

  $$(x_1 \vee x_4 \vee x_5) \wedge (x_2 \vee x_3 \vee x_4) \wedge (x_3 \vee x_5)$$

  is a monotone formula. It is easy to see that a monotone formula is always satisfiable — simply set each variable to 1. It is however natural to ask for a satisfying assignment with as few variables as possible set to 1. For example, in the above formula we can set two variables to 1 (say x_4 and x_3) and satisfy it. Motivated by this we consider the problem *Monotone SAT with Few True Variables*. The input to this problem is a monotone SAT formula ϕ and an integer k. The question is whether ϕ is satisfiable with at most k variables set to 1 (and the rest to 0). Prove that this problem is NP-Complete. *Hint*: Consider a reduction from Vertex Cover.

Problem 6. Consider the Partition problem. You are given integers $x_1, \ldots, x_n$, and you want to decide whether the numbers can be partitioned into two subsets S_1 and S_2 with the same sum:

$$\sum_{x_i \in S_1} x_i = \sum_{x_j \in S_2} x_j.$$

- Show that Partition is NP-Complete.

- Let T be the sum of the elements so that $T = \sum_{i=1}^{n} x_i$. Describe an algorithm for Partition that is polynomial in n and T.

- Why do the above not prove that P = NP?