Instructor: Lorna Stewart

CMPUT 304 Fall 2000 Final Exam

Date: Tues, Dec 12
Time allowed: 120 minutes
Total Marks: [80]

Question 1: [10] Let $X = x_1x_2 \ldots x_n$ and $Y = y_1y_2 \ldots y_m$ be two character strings. Give a $\Theta(nm)$ dynamic programming algorithm to find the longest substring common to X and Y. For example, the longest substring common to *photograph* and *tomography* is *ograph*. Be sure to describe the data structure, initialization, the method for computing, and the order in which to compute, the intermediate results, and the final answer.

Question 2: [10] P is a character string of length m consisting of letters and at most one asterisk ('*'). The asterisk is a "wild-card" character; it can match any sequence of zero or more characters. For example, if $P =$ 'sun*day' and $T =$ 'happysunshineday', there is a match beginning at 's' and ending at 'y'; the asterisk "matches" shine. Explain how you could use the KMP string matching algorithm to find such a match of P in a text string T (consisting of n letters; no asterisk), if there is one, and give an upper bound on the order of its worst-case complexity.

Question 3: [10] Several companies send representatives to a conference; the ith company sends m_i representatives. The organizers of the conference conduct simultaneous networking groups; the jth group can accomodate up to n_j participants. The organizers want to schedule all the participants into groups, but the participants from the same company must be in different groups. The groups need not all be filled.

Show how to use network flows to test whether the constraints can be satisfied. Clearly identify the flow network, the source and sink, the edge capacities, and describe how to use the network to test whether the constraints can be satisfied and, if so, find a schedule of participants into groups.

Question 4: [10]

(a) [3] Give a definition of NP in terms of Turing machines.

(b) [3] State Cook's theorem.

(c) [4] Briefly (2-3 sentences) describe the idea behind the proof of Cook's theorem.

Question 5: [10] Describe the error in the following fallacious "proof" that $P \neq NP$. Consider an algorithm for SAT: "On input ϕ, try all possible truth assignments to the variables. Accept if any assignment satisfies ϕ." This algorithm clearly requires exponential time. Thus SAT has exponential time complexity. Therefore SAT is not in P. Because SAT is in NP, it must be true that P is not equal to NP.

Question 6: [10] The *subgraph-isomorphism problem* takes two graphs G_1 and G_2 and asks whether G_1 is a subgraph of G_2. Show that the subgaph-isomorphism problem is NP-complete. Hint: Use a reduction from HAM-CYCLE.

Please turn over ...

Question 7: [10] The following results are proved in the textbook:

- CLIQUE $\leq_P$ VERTEX-COVER (p. 950)
- There exists an approximation algorithm for the vertex-cover problem with a ratio bound of 2 (p. 968)

Do these two results together imply that there is an approximation algorithm with constant ratio bound for the clique problem? Justify your answer. Use an example to illustrate your point.

Question 8: [10] Indicate, for each phrase below, whether it is true or false **for probabilistic algorithms**.

- may give answers that are not necessarily exact
- used for optimization problems only (not decision problems)
- may fail to give any answer at all
- returns answers that are guaranteed to be close to the correct answer
- may behave differently when applied twice to the same instance
- for a nonempty subset of instances, the correct answer is always returned
- makes random (or pseudorandom) choices during execution
- has bounded probability of returning a wrong answer or no answer, regardless of the input
- used for problems that are known to be NP-complete or NP-hard only
- finds a correct solution with probability greater than a certain threshold, for every instance