Two hours

UNIVERSITY OF MANCHESTER
SCHOOL OF COMPUTER SCIENCE

Advanced Algorithms I

Date: Thursday 23rd January 2014
Time: 09:45 - 11:45

Please answer any THREE Questions from the FIVE Questions provided

This is a CLOSED book examination

The use of electronic calculators is NOT permitted
1. Stable marriages.

   a) Let $B$ and $G$ be finite, non-empty sets of the same cardinality. Assume that each element of $B$ is associated with a total preference ordering for the elements of $G$, and each element of $G$ is associated with a total preference ordering for the elements of $B$. What is meant by a stable matching between $B$ and $G$? (2 marks)

   b) Describe the Gale-Shapley algorithm for generating stable matchings. (6 marks)

   c) Let $B = \{b_1, b_2, b_3\} \text{ and } G = \{g_1, g_2, g_3\}$. For each $i \ (0 \leq i < 3)$, let the preferences of $b_i$ be given by $g_{i+1} > g_i > g_{i-1}$ (where arithmetic in subscripts is modulo 3); and similarly, for each $j \ (0 \leq j < 3)$, let the preferences of $g_j$ be given by $b_{j+1} > b_j > b_{j-1}$. What matching will the Gale-Shapley algorithm produce in this case? (Assume that, as usual, the elements of $B$ do the ‘proposing’.) Why is this not good news for the elements of $G$? (4 marks)

   d) Give an example of a stable matching with equitable outcomes for $B$ and $G$. Demonstrate that your matching really is stable. (4 marks)

   e) It is sometimes said that the Gale-Shapley algorithm produces stable matchings that are optimal for the elements of $B$. Explain what this means. Why is there only one such matching? (4 marks)

a) Define a flow network. In the context of a flow network, what is a flow, and what is the value of that flow? (4 marks)

b) Let $G = (U, V, E)$ be a bipartite graph. What is a perfect matching for $G$? (2 marks)

c) Explain how, given a bipartite graph $G = (U, V, E)$ with $|U| = |V| = n$, we can construct a flow network with the property that the maximal integral-valued flows for that network with value $n$ correspond exactly to the perfect matchings for $G$. (You should use a diagram in your explanation.) (4 marks)

d) Let $U = \{u_1, u_2\}$, $V = \{v_1, v_2\}$ and $E = \{(u_1, v_1), (u_1, v_2), (u_2, v_1)\}$. Draw the corresponding flow network $N$ in this case. (4 marks)

e) Let $f_0$ be the zero flow for this network (i.e. flow of 0 along every edge). Draw the auxiliary directed graph $N_{f_0}$, and clearly highlight an augmenting path in this network through the vertices corresponding to $u_1$ and $v_1$. (2 marks)

f) Augment $f_0$ along this path, and let the new flow be $f_1$. Draw the auxiliary directed graph $N_{f_1}$, and clearly highlight an augmenting path in this network. (2 marks)

g) Let the new flow be $f_2$. Show how a perfect matching for $G$ can be extracted from $f_2$. (2 marks)
3. Linear programming.

a) Maximize \( x + y \) subject to the constraints

\[
\begin{align*}
y &\leq 13 - \frac{3}{10}x \\
x &\leq 15 - \frac{1}{2}y \\
x &\geq 0 \\
y &\geq 0.
\end{align*}
\]

(12 marks)

b) Suppose we are given a linear programming problem in the form

\[
\begin{align*}
\text{maximize} & \quad c^T x \\
\text{subject to} & \quad Ax \leq b,
\end{align*}
\]

where the variables are allowed to range over \( \mathbb{R} \). Show that this problem may be transformed into one in standard form:

\[
\begin{align*}
\text{maximize} & \quad (c')^T w \\
\text{subject to} & \quad A'w \leq b' \\
& \quad w \geq 0,
\end{align*}
\]

and give expressions for \( A' \) and \( c' \). (4 marks)

c) Let \( G = (V, E) \) be a directed graph, \( s, t \) vertices of \( G \), and \( c \) a function \( c : E \to \mathbb{R}^+ \). Express the problem of maximizing flow in the flow network \( (V, E, s, t, c) \) as a linear programming problem. (4 marks)
4. Reductions.
   
   a) Explain what is meant by a *many-one polynomial time reduction*. (4 marks)
   
   b) Define the problems SAT and $k$-SAT, for $k$ a positive integer. (4 marks)
   
   c) Give a many-one polynomial time reduction from SAT to 3-SAT, showing the correctness of your reduction. (4 marks)
   
   d) Define the problem INTEGER LINEAR PROGRAMMING FEASIBILITY. (4 marks)
   
   e) Give a many-one polynomial time reduction from SAT to INTEGER LINEAR PROGRAMMING FEASIBILITY, showing the correctness of your reduction, and deduce a lower complexity bound for INTEGER LINEAR PROGRAMMING FEASIBILITY. (4 marks)
5. Space complexity.

a) State the Immerman-Szelepcsényi theorem. (4 marks)

b) Define the problem DIRECTED GRAPH REACHABILITY. For what familiar complexity class is this problem known to be complete? (4 marks)

c) Let $S$ be the set of propositional formulas of the forms

\[ p \]
\[ \neg p \]
\[ p \rightarrow q \]

where $p$ and $q$ are proposition letters. Define the problem $S$-SAT as follows:

Given: a finite set $\Phi$ of formulas of $S$.
Output: Y if $\Phi$ is satisfiable, N otherwise.

For any finite set $\Phi$ of formulas of $S$, define the directed graph $G_{\Phi} = (V_{\Phi}, E_{\Phi})$ by taking $V_{\Phi}$ to be the set of proposition letters mentioned in $\Phi$, and $E$ the set $\{(p, q) \mid p \rightarrow q \in \Phi\}$. Show carefully that $\Phi$ is satisfiable if and only if there do not exist formulas $p \in \Phi$ and $\neg q \in \Phi$ such that $q$ is reachable from $p$ in $V_{\Phi}$. (8 marks)

d) What can you conclude from the above facts about the complexity of $S$-SAT? Explain your reasoning. (4 marks)