Two hours

UNIVERSITY OF MANCHESTER
SCHOOL OF COMPUTER SCIENCE

Advanced Algorithms I

Date:   Thursday 17th January 2013
Time:   14:00 - 16:00

Please answer exactly TWO Questions from Section A
and exactly TWO Questions from Section B

Use a SEPARATE answerbook for each SECTION.

This is a CLOSED book examination
The use of electronic calculators is NOT permitted

[PTO]
Section A

Answer two and only two questions from this section

1. a) What is a strongly connected component of a directed graph? (2 marks)

b) Describe an algorithm for the depth-first search of directed graphs. (4 marks)

c) How does Tarjan’s Algorithm exploit depth-first search to find strongly connected components? Your answer should include the implementation of LowLink and explain its significance for the efficient implementation of Tarjan’s Algorithm. (6 marks)

d) Analyze the complexity of Tarjan’s Algorithm to show that it is linear in both edges and vertices. Explain the significance of LowLink in your calculations. (8 marks)

2. a) Explain how to solve linear programming problems graphically in two dimensions. You should explain any restrictions on the system you are solving. (2 marks)

b) Give a generalized description of the problem in matrix form for higher dimensional problems. (2 marks)

c) Give an algorithm for the Simplex Method. (4 marks)

d) Apply the Simplex Algorithm to the following problem.

Maximize \( f = 2x + 3y + z \)
subject to \( 3x + 6y + z \leq 6 \)
\( 4x + 2y + z \leq 4 \)
\( x - y + z \leq 3 \)
and \( x \geq 0, y \geq 0, z \geq 0 \)

(6 marks)

e) How can the algorithm be adjusted if some of the parameters are restricted to be integers? Illustrate your answer by solving the previous problem with \( x, y \) and \( z \) restricted to integer values. (6 marks)
3. a) Why is parallel programming so difficult when compared to sequential programming? (6 marks)

b) What can we do in practice to make parallel programming more straightforward? (6 marks)

c) Consider the following $\pi$-calculus description of Hoare’s Vending Machine. Why is it unlikely to be what the customer wants? Give a corrected form of this machine. (8 marks)
Section B

Answer two and only two questions from this section

4. a) If $f : \mathbb{N} \rightarrow \mathbb{N}$ is a function, explain the terms (i) $\text{SPACE}(f)$, (ii) $\text{NSPACE}(f)$. (2 marks)

b) The problem REACHABILITY is defined as follows.

Given A directed graph $G$ and nodes $u, v$ of $G$.

Output Yes, if there is a path in $G$ from $u$ to $v$; No, otherwise.

Using pseudocode, give an algorithm showing that REACHABILITY is in $\text{SPACE}(\log^2 n)$. (4 marks)

c) If $M$ is a Turing machine and $x$ a string over the alphabet of $M$, what is the configuration graph for $M$ with input $x$? You may give your answer informally, with the aid of a diagram; however, you should explain precisely what the vertices and edges of the configuration graph represent. (5 marks)

d) Using the notion of a configuration graph, explain why, for “reasonable” functions $f(n) \geq \log n$, $\text{NSPACE}(f) \subseteq \text{SPACE}(f^2)$. (5 marks)

e) Explain why the result in Question 4a shows that $\text{PSPACE} = \text{NPSPACE}$, but not $\text{LOGSPACE} = \text{NLOGPSpace}$. (4 marks)

5. a) Explain the terms:

i) NPTime

ii) logarithmic space many-one reduction;

iii) NPTime-hard (with respect to logarithmic space many-one reductions);

iv) NPTime-complete (with respect to logarithmic space many-one reductions). (6 marks)
b) In the context of graph algorithms, define the problem HAMILTONIAN CIRCUIT.

(4 marks)

c) What is meant by the travelling salesman problem? Explain why, and in precisely what sense, this problem can be said to be in NPTime.

(4 marks)

d) Give a reduction of HAMILTONIAN CIRCUIT to (a suitable version of) the travelling salesman problem, and deduce that the latter is NPTime-complete.

(6 marks)

6. a) Explain how a Turing machine (over some alphabet Σ) can be represented as a string (over some larger alphabet Σ' ⊇ Σ).

(4 marks)

b) Explain the terms:

i) recursively enumerable;

ii) recursive.

(2 marks)

c) Is the set of strings representing Turing machines given in your answer to Q.6a recursively enumerable, or even recursive? Explain your answer informally.

(2 marks)

d) Explain the expressions (i) partial function N → N; (ii) total function N → N;

(2 marks)

e) Explain the term universal Turing machine.

(2 marks)
f) Let $M$ be a Turing machine which outputs on its tape an infinite sequence of strings representing a sequence of Turing machines $M_1, M_2, \ldots$, with the property that each $M_i$ computes a total function $f_i : \mathbb{N} \to \mathbb{N}$. Construct a Turing machine $M^*$ which computes a total function $f^* : \mathbb{N} \to \mathbb{N}$ guaranteed to be different from every $f_i$. You may describe $M^*$ informally, for example using a flowchart if you find this easier; however, you must be clear as to how it computes its output.

(4 marks)

g) Hence show that the set of strings describing Turing machines over the alphabet \{0, 1\} that compute total functions $\mathbb{N} \to \mathbb{N}$ is not recursively enumerable.

(4 marks)