CS_232 2004/05
Algorithms and Complexity

(Attempt 2 questions out of 3)

Question 1 Cryptology

(a) Modular arithmetic and the Euclidean algorithm

(i) Compute $5 +_{10} 7 \in \mathbb{Z}_{10}$, $5 -_9 8 \in \mathbb{Z}_9$ and $3 *_8 9 \in \mathbb{Z}_8$, showing your computations.

[3 marks]

(ii) Compute $\mathrm{pow}_{21}(3, 99) \in \mathbb{Z}_{21}$, showing your computations.

[3 marks]

(iii) Compute $\gcd(108, 140)$ with the Euclidean algorithm, showing the Euclidean sequence.

[2 marks]

(iv) Extend the computation of $\gcd(108, 140)$ with the computation of the Euclidean extension sequence, and derive coefficients $x, y \in \mathbb{Z}$ with $x \cdot 108 + y \cdot 140 = \gcd(108, 140)$.

[2 marks]

(v) Decide whether 108 is invertible in $\mathbb{Z}_{140}$ and whether 99 is invertible in $\mathbb{Z}_{140}$; in the affirmative case compute the inverse (showing your computations).

[2 marks]

(b) RSA

(i) Encrypt the plaintext $m = 12 \in \mathbb{Z}_{143}$ as $\mathrm{RSA}_{(143,49)}(12)$, using the public key $n = 143$, $e = 49$ (show all details of your computation).

[3 marks]

(ii) Using the secret information $143 = 11 * 13$, compute the secret key d corresponding to $e = 49$ and decrypt the ciphertext $c = 48$ as $\mathrm{RSA}^{-1}_{(143,49)}(48)$ (show all details of your computation).

[3 marks]

(iii) Comment on the above choice of the secret prime numbers.

[2 marks]

(c) Explain the idea of the zero knowledge proof for 3-colourability.

[5 marks]

Question 2 Graphs and graph traversal

(a) Fundamental properties

(i) Define "graphs" and "general graphs".

[2 marks]

(ii) Explain the notion of connected components of a graph, and how they can be computed efficiently. What is the time complexity of your algorithm?

[3 marks]

(iii) Explain the notion of a spanning tree of a connected graph, and how one can be found efficiently. What is the time complexity of your algorithm?

[3 marks]

(b) BFS and DFS

(i) What is the characteristic property of BFS spanning trees of (connected) graphs?

[2 marks]

(ii) Try to develop a characterisation of DFS spanning trees of (connected) graphs (what makes DFS trees special?).

[4 marks]

(c) The `graph_traversal` procedure

(i) Describe *in your own words* the idea of the `graph_traversal` procedure (do not give the pseudo code here, but show your understanding of the principles).

[4 marks]

(ii) Given a finite connected graph G, how many tree edges will the procedure `graph_traversal` discover (for any buffer strategy)? Give reasons for your answer.

[3 marks]

(iii) Describe how `graph_traversal` can be specialised to yield Dijkstra's algorithm for computing an SPT for a graph with non-negative edge weights.

[4 marks]

Question 3 Complexity theory

(a) Complexity classes

(i) Define the complexity classes P, NP and co-NP, and explain the notion of NP-completeness.

[5 marks]

(ii) State for the following problems whether they are in P, in NP or whether they are NP-complete (choose in each case the most precise answer):

1. Is a number a prime number?
2. Given a graph with rational edge weights, two vertices of the graph and a rational number b, is there a path P connecting the two vertices such that P has a weight less than or equal to b?
3. Is a propositional formula in conjunctive normal form satisfiable?
4. Is a graph bipartite?
5. Given natural numbers $n_1, \ldots, n_k$ and t, is it possible to find a subset of $\{n_1, \ldots, n_k\}$ such that the sum of the elements of this subset is exactly t?
6. Given a composite number, find a non-trivial factor.

[3 marks]

(iii) State the basic idea to show that exponential time algorithms can decide strictly more problems than polynomial time algorithms.

[3 marks]

(b) The SAT problem

(i) Define the SAT problem.

[2 marks]

(ii) Decide for each of the following clause-sets F_i, $i = 1, 2$, whether F_i is satisfiable or not, and justify your answers:

$$F_1 := \{\, \{a, b\}, \{\overline{a}, b\}, \{\overline{b}, c, d\}, \{\overline{b}, \overline{c}, d\}, \{\overline{b}, \overline{d}\} \,\}$$
$$F_2 := \{\, \{a, b, c\}, \{\overline{a}, b, c\}, \{a, \overline{b}, c\}, \{a, b, \overline{c}\}, \{\overline{a}, \overline{b}, c\}, \{\overline{a}, b, \overline{c}\}, \{a, \overline{b}, \overline{c}\} \,\}.$$

[4 marks]

(iii) Describe how the k-colouring problem can be reduced in polynomial time to the k-SAT problem.

[4 marks]

(iv) Describe the backtracking approach to solve the SAT problem, and what are the advantages of this approach, compared with the simple enumeration of all possible solutions.

[4 marks]