PRIFYSGOL CYMRU; UNIVERSITY OF WALES

DEGREE EXAMINATIONS JANUARY 2003

SWANSEA

Computer Science

CS 232 Algorithms and Complexity

Attempt 2 questions out of 3

Time allowed: 2 hours

Students are permitted to use the dictionaries provided by the University

Students are NOT permitted to use calculators

CS 232
ALGORITHMS AND COMPLEXITY
January 2003

(*Attempt 2 questions out of 3*)

Question 1

(a) What is the appropriate notion of input size for a numerical algorithm that computes some function over the natural numbers?
Define what it means for such a numerical function to be of linear or polynomial time complexity.

Classify the following simple numerical functions according to whether or not they are of polynomial, or even linear, time complexity:

$$
\begin{array}{lcll}
(a, b) & \mapsto & \gcd(a, b) & \text{(greatest common divisor)} \\
(a, b) & \mapsto & a^b & \text{(exponentiation)}
\end{array}
$$

Justify your answers. **[7 marks]**

(b) (i) Give pseudocode for a feasible algorithm for modular exponentiation on the basis of repeated squaring and briefly explain its main features.

(ii) Apply modular exponentiation by repeated squaring to compute the value of $7^{555} \bmod 17$ without unnecessarily large intermediate results. Display all relevant intermediate results in a table.

[9 marks]

(c) (i) What does it mean for a decision problem $D \subseteq I$ to be in P?

(ii) Define in general terms what it means for a decision problem $D_1 \subseteq I_1$ to be polynomially reducible to a second decision problem $D_2 \subseteq I_2$, $D_1 \leqslant_{\text{poly}} D_2$ for short.
What would $D_1 \leqslant_{\text{poly}} D_2$ tell us
– about the complexity of D_1 if D_2 is in P?
– about the complexity of D_2 if D_1 is not in P?
– if D_1 was NP-complete and D_2 in P?
In each case, briefly justify your answer.

(iii) Sketch in broad outline the argument for 3-COL $\leqslant_{\text{poly}}$ 3-CNF-SAT.

[9 marks]

Question 2

(a) (i) What is the characteristic feature of traditional symmetric cryptographic schemes? Give an example.

(ii) Explain the main idea in the Diffie Hellman Merkle protocol for key generation, and explain how this addresses the problem raised by symmetric schemes.

(iii) Explain the advantage of public key systems over traditional symmetric systems.

[9 marks]

(b) Explain how RSA works, by outlining the process for generation of a key pair, and the use of these matching keys in encryption and decryption, in the following example:

(i) Alice selects primes $p = 23$, $q = 5$, and chooses $e = 7$ as the exponent for her public key; explain how Alice generates a matching secret key. Compute Alice's secret key, indicating all essential itermediate results in the use of ExtendedEuclid for this.

(ii) Bob wants to send the message $M = 3$; compute the result of encrypting M with Alice's public key $(115, 7)$, indicating all the relevant intermediate results in the feasible modular exponentiation procedure.

(iii) State which decryption transformation Alice has to apply to the encrypted message from (ii).

[10 marks]

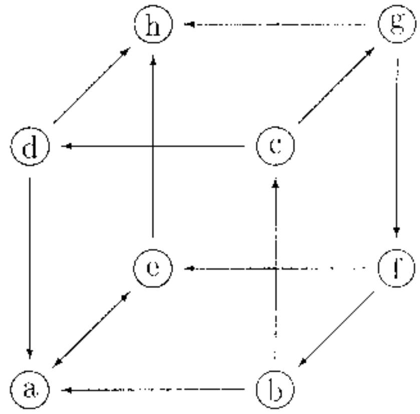
(c) Briefly explain

(i) the complexity considerations that make RSA key generation feasible.

(ii) the complexity assumptions for the security of Diffie Hellman Merkle key generation.

(iii) the complexity assumptions for the security of RSA encryption.

[6 marks]



G_1

Question 3

(a) Consider the graph G_1 depicted to the right.

(i) Give its adjacency list representation with respect to the alphabetical enumeration of vertices.

(ii) Describe the run of DFS on G_1 in the representation from (i):
– indicate the order in which vertices are discovered.
– display the resulting DFS forest and time stamps t_1/t_2.
– display the parenthesis structure of the nested DFS-VISIT calls.

[**9 marks**]

(b) (i) Define the notion of strongly connected components and indicate the strongly connected components of G_1.

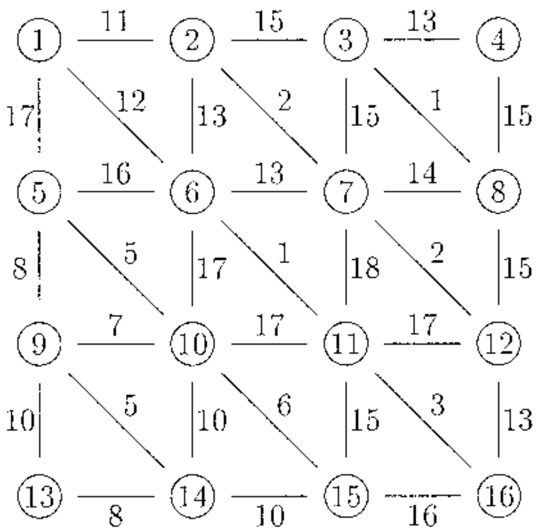
(ii) Working with the result from the DFS-pass on G_1 obtained in part (a), follow the further stages in the SCC algorithm based on a second DFS-pass on the transpose G_1^T of G_1 with the appropriate enumeration of vertices.

[**8 marks**]

(c) (i) Explain in words Kruskal's strategy for finding a minimal spanning tree for a weighted undirected graph. Follow Kruskal's strategy on the undirected graph G_2 with weights ω indicated below: list a sequence in which edges may be selected and determine a minimal spanning tree.

(ii) Explain in brief outline the key argument showing that Kruskal's greedy strategy does guarantee that a minimal spanning tree is found.

[**8 marks**]



(G_2, ω)