OLLSCOIL NA hÉIREANN, MÁ NUAD

NATIONAL UNIVERSITY OF IRELAND, MAYNOOTH

M.SC. IN SOFTWARE ENGINEERING EXAMINATION

HIGHER DIPLOMA IN SOFTWARE ENGINEERING EXAMINATION

DECEMBER 2001

COMPUTER SCIENCE

PAPER CS605

MATHEMATICS AND THEORY OF COMPUTER SCIENCE

Dr. R. Procter, Mr. S. Brown, Dr. J. P. Gibson, Mr. T. Naughton

Attempt any FOUR questions. Time Allowed: 3 hours.

1. (a) State whether each of the following is true or false. [6 marks]

   i. $\emptyset \in \emptyset$

   ii. $\emptyset = 2^{\emptyset}$

   iii. $\{a, b\} \subseteq 2^{\{a,b,\{a,b\}\}}$

   (b) Let $\Sigma = \{a, b, c\}$ and let $L = \{w : w \in \Sigma^*\}$. Write down the first five elements in the lexicographical ordering of L, where Σ has the usual alphabetical ordering (a, b, c). [5 marks]

   (c) Let $\mathbb{R}$ be the set of real numbers. Let $\mathbb{Z}$ be the set of integers. Let $\mathbb{N}$ be the set of natural numbers. State whether each of the following sets is countable or not. [8 marks]

   i. $2^{\mathbb{R}}$

   ii. $2^{\mathbb{Z}}$

   iii. $2^{\mathbb{N}}$

   iv. $2^{\mathbb{R}\cap\mathbb{Z}}$

   (d) Can you enumerate the set of all words over a finite alphabet? Prove your answer. [6 marks]

2. Consider the following nondeterministic two-stack pushdown automaton $M = (Q, \Sigma, \Delta, q_0, F) = (\{00\}, \{a, b, c\}, \Delta, 00, \{00\})$ where Δ is given by

q	s	pop	q'	psh
00	a	$(-,-)$	00	$(a,-)$
00	b	$(-,-)$	00	$(b,-)$
00	c	$(-,-)$	00	$(c,-)$
00	$-$	$(a,-)$	00	$(-,a)$
00	$-$	$(b,-)$	00	$(-,b)$
00	$-$	$(c,-)$	00	$(-,c)$
00	$-$	$(-,a)$	00	$(a,-)$
00	$-$	$(-,b)$	00	$(b,-)$
00	$-$	$(-,c)$	00	$(c,-)$
00	a	(b,c)	00	$(-,-)$
00	a	(c,b)	00	$(-,-)$
00	b	(a,c)	00	$(-,-)$
00	b	(c,a)	00	$(-,-)$
00	c	(a,b)	00	$(-,-)$
00	c	(b,a)	00	$(-,-)$

and where symbol '$-$' replaces 'e' simply to make the table easier to read.

(a) What does M do? [10 marks]

(b) Construct a nondeterministic two-stack pushdown automaton to accept the language $L = \{w : w \in \{a, b, c\}^*, w = a^n b^n c^n, n \in \mathbb{N}, n \geq 0\}$. [10 marks]

(c) State precisely what nondeterministic two-stack pushdown automata are capable of and not capable of. [5 marks]

3. You are given a two-tape Turing machine $T = (Q, \Sigma, I, q_0, F) = (\{00, 01, 02, 03\}, \{a, b, -\}, I, 00, \{03\})$. The head of the first tape will be positioned at the beginning of the input and the second tape will be blank. I is

q	s	q'	s'	m
00	$(a,-)$	00	$(a,-)$	(R,S)
00	$(b,-)$	00	$(b,-)$	(R,S)
00	$(-,-)$	01	$(-,-)$	(L,S)
01	$(a,-)$	01	$(-,-)$	(L,S)
01	$(b,-)$	02	$(-,b)$	(L,R)
01	$(-,-)$	03	$(-,-)$	(R,S)
02	$(a,-)$	01	(a,b)	(S,R)
02	$(b,-)$	01	(b,b)	(S,R)
02	$(-,-)$	01	$(-,b)$	(S,R)

(a) What does T do? Give as concise an explanation as you can. [8 marks]

(b) Convert T into a functionally-identical Turing machine that requires at least one less state than T. [8 marks]

(c) Explain, with examples as necessary, the following terms: *accept*, *decide*, and *recursively enumerable*. [9 marks]

4. A software company wants to improve the efficiency of function call parameter passing in its software products. In the existing code, there were many instances of objects being passed by value to functions that did not change these objects under any circumstances. In order to avoid the overhead of implicitly copying the objects when such functions are called it would be preferable to pass the objects by reference. The company are writing a routine that takes a description of a function $f()$ as input and determines whether its arguments (currently passed by value) would be updated during the execution of $f()$ or not. If an argument would not be updated then it could be safely passed by reference without changing the semantics of the program. (Assume that functions only refer to variables passed as arguments or declared within the function body — i.e. there are no global variables to contend with). [25 marks]

   (a) Consider a restricted version of the problem where $f()$ contains only one argument A and there are no other function calls within the body of $f()$. Prove the decidability or undecidability of this problem.

   (b) Consider the more general case where $f()$ contains an arbitrary number of arguments ($A, B, C, \ldots$ and so on) and $f()$ can contain calls to itself or other functions. Prove the decidability or undecidability of this problem.

5. Consider the following sextuple of language classes (regular, context-free, recursively enumerable, recursive, $\mathcal{P}, \mathcal{NP}$). Every language can be associated with a binary sextuple where symbol 1 denotes membership and 0 denotes nonmembership of the class in question. For example, if a language was in the first class and not in any of the others, it would be associated with the binary sextuple $(1, 0, 0, 0, 0, 0)$.

   (a) State the binary sextuple associated with each of the following languages.

      i. The language $L = \{a, aa, ab\}$. [1 marks]

      ii. The language $L = \{w : w \in \{a, b\}^*, |w| \text{ is odd}, w \text{ has exactly one } b \text{ positioned exactly in the centre of the word}\}$. [3 marks]

      iii. The language $L = \{w : w \in \{a, b\}^*, |w| \text{ is odd}, w \text{ has exactly one } b\}$. [3 marks]

      iv. The language of chess board configurations for which white can win. [3 marks]

      v. The language of Turing machines that contain a state 01. [3 marks]

      vi. The language of Turing machines that go into a state 01. [3 marks]

      vii. The language of set systems that can be hit by a set of cardinality 12. [3 marks]

      viii. The language of graphs that have a tour that visits each vertex at least once. [3 marks]

   (b) Finally, state three binary sextuples that cannot exist, and indicate where the contradiction occurs in each case. [3 marks]

6. (a) What is a polynomially-bounded (or polynomial-time) reduction? Illustrate with an example why these are of interest in computational complexity theory. [5 marks]

   (b) You want to find an algorithmic solution for problem A. You know that A is in $\mathcal{NP}$. Should you look for an efficient algorithm for A? Explain. [4 marks]

   (c) Is the problem of writing out the factorial of a number in unary $\mathcal{NP}$-complete or $\mathcal{NP}$-hard (e.g. $n! = 111111$ for $n = 3$)? Explain. [4 marks]

   (d) Finland's state-run off-licence chain, Alko, is facing competition for the first time and it decides to lower the price of selected bottles of spirits. Unfortunately, while the larger Alko outlets sell each of the m brands, some of the smaller outlets sell but a limited range. Alko wants to discount no more than k possible brands such that each of its n outlets stocks at least one discounted item. "Surely a computer this powerful can figure it out", one of the board was heard to say. Prove that this problem is $\mathcal{NP}$-complete. You are given only that SATISFIABILITY is $\mathcal{NP}$-complete. [12 marks]