UNIVERSITY OF EDINBURGH

COLLEGE OF SCIENCE AND ENGINEERING

SCHOOL OF INFORMATICS

INFORMATICS 1 - COMPUTATION AND LOGIC

Monday 11 th August 2014

14:30 to 16:30

INSTRUCTIONS TO CANDIDATES

1. **Note that ALL QUESTIONS ARE COMPULSORY.**
2. **DIFFERENT QUESTIONS MAY HAVE DIFFERENT NUMBERS OF TOTAL MARKS. Take note of this in allocating time to questions.**
3. **Calculators may not be used in this examination.**

Convener: J. Bradfield
External Examiner: C. Johnson

THIS EXAMINATION WILL BE MARKED ANONYMOUSLY

1. In a certain imaginary land, there are three kinds of inhabitants: knights, knaves and spies. Knights always tell the truth; knaves always lie; spies sometimes tell the truth and sometimes lie.

   Consider the following statements made by three inhabitants of the land:

   A: "I am Smiley."

   B: "What A says is true."

   C: "I am Smiley."

   Of these three inhabitants, one is a knight, one is a knave and one is a spy. The spy is the only one named Smiley. Which of A, B and C is the spy? *[10 marks]*

2. Show that the following logical expressions are equivalent, using the method of truth tables:

   (a) $p \to q$; $p \to (p \to (p \to q))$ *[4 marks]*

   (b) $(p \text{ or } q) \text{ and } (not(p) \text{ or } r)$; $(p \text{ and } r) \text{ or } (q \text{ and } not(p)) \text{ or } (q \text{ and } r)$ *[5 marks]*

   (c) $((p \leftrightarrow q) \text{ and } (p \leftrightarrow r))$; $(p \text{ and } q \text{ and } r) \text{ or } (not(p) \text{ and } not(q) \text{ and } not(r))$ *[6 marks]*

3. (a) Explain what it means for a formula to be in conjunctive normal form (CNF). *[3 marks]*

   (b) Convert the following propositional expressions to CNF:

      i. $(p \to q) \to (q \to r)$ *[5 marks]*

      ii. $not((p \text{ or } q) \text{ and } (not(p) \text{ or } not(q)))$ *[5 marks]*

      iii. $p \text{ and } not(p \to q)$ *[3 marks]*

   (c) Show that the following set of clauses is consistent, using the Davis-Putnam algorithm for resolution: *[9 marks]*

   $$[[a, not(a), b], [a, not(c), d], [b, c, not(d)], [c, not(d)], [not(b), not(d)], [a, c]]$$

4. (a) Explain the notions of finite-state transducer and finite-state acceptor, with an emphasis on the differences between the notions. *[4 marks]*

   (b) Design a finite-state machine over the alphabet $\{0, 1\}$ which accepts all strings which have length a multiple of 4 and which have an odd number of 1s. *[9 marks]*

5. Give a formal proof that there is no finite-state machine over the alphabet $\{1\}$ which accepts only those strings with length a power of 2. *[15 marks]*

6. Consider the following formal description of a non-deterministic finite state machine M: $M = (Q, \Sigma, s_0, F, \delta)$, where
$Q = \{1, 2, 3, 4, 5, 6, 7, 8\}$
$\Sigma = \{1, 2\}$
$s_0 = \{1\}$
$F = \{8\}$
$\delta = \{(1,1,2), (1,1,4), (2,2,2), (2,2,3), (3,1,3), (3,\epsilon,8), (4,1,5), (5,1,5), (5,2,5), (4,2,6), (6,1,4), (6,1,6), (6,2,7), (7,\epsilon,8)\}$.

(a) Draw the machine M. [*2 marks*]

(b) Write down a regular expression for the language accepted by M. [*10 marks*]

(c) Give a *determistic* finite-state machine equivalent to M. [*10 marks*]