NUI MAYNOOTH

Ollscoil na hÉireann Má Nuad

OLLSCOIL NA hÉIREANN MÁ NUAD

NATIONAL UNIVERSITY OF IRELAND MAYNOOTH

M.Sc. in Software Engineering Examination

SEMESTER 1
2005-2006

MATHEMATICS AND THEORY OF COMPUTER SCIENCE

PAPER CS605

Dr. Ita Richardson, Prof. Ronan Reilly, Mr. Tom Naughton

Time allowed: 3 hours

Answer ***three*** questions

All questions carry equal marks

Additional material allowed:
One copy of M. Sipser, *Introduction to the Theory of Computation* (PWS, Boston, 1997), containing no annotations or extra pages AND/OR one copy of M. Sipser, *Introduction to the Theory of Computation – Second Edition* (Thomson, Boston, 2005), containing no annotations or extra pages. Please see declaration page at back.

1. (a) Figure 1 on page 3 illustrates the space of languages 2^{Σ^*} for some finite Σ, where $|\Sigma| > 1$. Place each of the following languages, **and** its complement, in its appropriate place in this space. [10 marks]

   i. $\text{MEM}_{\text{J}} = \{\langle J_1, J_2\rangle : J_1$ and J_2 are Java functions without arguments that require the same minimum amount of memory to run without crashing$\}$

   ii. $\text{AT}_{\text{TM}} = \{\langle M, w\rangle : M$ is a Turing machine and w is a word and M accepts $w\}$

   iii. $\text{NEQ}_{\text{J}} = \{\langle J_1, J_2\rangle : J_1$ and J_2 are Java functions that recognise different languages$\}$

   iv. $\text{SON}_{\text{TM}} = \{S : S$ is a binary string (strings are finite by default) and S contains at least three '1's$\}$

   v. $\text{SD}_{\text{J}} = \{\langle J_1, J_2\rangle : J_1$ and J_2 are Java functions that contain declaration statements for an integer called $a\}$

   (b) Prove that the set of all languages over a finite alphabet is uncountable. [10 marks]

   (c) Let the HITTINGSET problem be defined as follows. Given a finite set of subsets $A_1, A_2, \ldots, A_n$ of a finite set A, is there another subset of A containing at most k elements that has a nonempty intersection with each $A_1, A_2, \ldots, A_n$? Prove that HITTINGSET is in $\mathcal{NP}$. [5 marks]

2. (a) Let the language $\text{VAREQ}_{\text{C++}}$ be defined as $\text{VAREQ}_{\text{C++}} = \{\langle C, a, b\rangle : C$ is a C++ program, a and b are integer variables declared in C, and when C is run a and b have the same value at least once$\}$. You are given that $\text{HALT}_{\text{C++}}$ is undecidable. $\text{HALT}_{\text{C++}}$ is defined as $\text{HALT}_{\text{C++}} = \{\langle C, w\rangle : C$ is a C++ function, and C halts on its string input $w\}$. [12 marks]

   Prove that $\text{VAREQ}_{\text{C++}}$ is undecidable. You may answer this question by assigning a name, mathematical construct, or piece of pseudocode to each of the numbered blanks in the proof template in Figure 2 on page 3. Where blanks have the same number, this denotes their contents will be the same. Alternatively, you can choose to ignore the template and construct your own proof from scratch.

   (b) Prove that $\text{VAREQ}_{\text{C++}}$ is Turing-recognisable or prove that it is not Turing-recognisable. [4 marks]

   (c) Give a definition of the language $\overline{\text{VAREQ}_{\text{C++}}}$ (the complement of $\text{VAREQ}_{\text{C++}}$). Prove that $\overline{\text{VAREQ}_{\text{C++}}}$ is Turing-recognisable or prove that it is not Turing-recognisable. [4 marks]

   (d) Let $L = \{w : w \in \{a, b\}^*, w$ contains the substring baa but does not contain it at the beginning of the word$\}$. Construct (i) an finite automaton to recognize L, and (ii) a regular expression that evaluates to L. [5 marks]

3. (a) For each of the following languages, prove that it is regular or prove that it is not regular. [20 marks]

   i. $L_0 = \{uv : u \in \{0,1\}^*, v \in \{0\}^*, |u| = |v|\}$
   ii. $L_1 = \{uv : u \in \{0,1\}^*, v \in \{0,1\}^*, |u| = |v|\}$
   iii. $L_2 = \{w : w \in \{0,1\}^*, |w|$ is odd and the middle symbol of w is 0$\}$
   iv. $L_3 = \emptyset$
   v. $L_4 = \{uv : u, v \in \{a,b\}^*, |u| = |v|, u \neq v\}$

   (b) Prove that every nondeterministic finite automaton can be converted into an equivalent one that has a single accept state. Your proof should consist of an unambiguous sequence of steps that describes how to construct the equivalent machine. [5 marks]

4. (a) The following partial Turing machine M claims to recognise the language $L = \{w1w : w \in \{a,b\}^*\}$. However, four rows are missing that stop this Turing machine from working correctly. The start state is 00. To accept a word, M goes into state 99. The symbol '–' denotes a blank. [8 marks]

S_i	R	S_f	W	M
00	1	98	1	R
98	–	99	–	R
00	a	01	–	R
00	b	02	–	R
01	a	01	a	R
01	b	01	b	R
01	1	03	1	R
02	a	02	a	R
02	b	02	b	R
03	a	05	x	L
04	x	04	x	R
04	b	05	x	L
05	a	05	a	L
05	b	05	b	L
05	x	05	x	L
05	–	00	–	R

   State the four missing rows.

   (b) Specify new rows and new states (if necessary) that will convert this recogniser Turing machine M for L into a decider Turing machine M' for L. [4 marks]

   (c) In the worst case, for an input word of length N, what is the maximum number of timesteps required for M' to decide the input word, and the maximum number of tape cells written to by M' to decide the word. Calculate these complexities exactly. [4 marks]

   (d) Let $L = \{uv : u, v \in \{a,b\}^*, v$ contains an equal number of as and bs$\}$.

   i. Construct a pushdown automaton (give the full state diagram) to accept L. [4 marks]
   ii. Give a context-free grammar that generates L. [3 marks]
   iii. Can L be accepted by a deterministic pushdown automaton? Justify your answer. [2 marks]

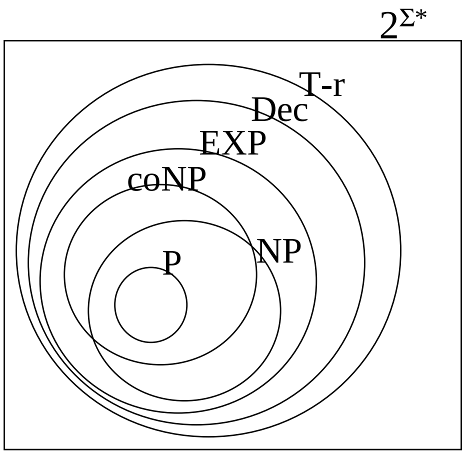

Figure 1: Illustration of the space of languages for question 1a: EXP denotes EXPTIME, Dec denotes the decidable languages (sometimes called the recursive languages), and T-r denotes the Turing-recognisable languages (sometimes called the recursively enumerable languages).

Proof. We will use a mapping reduction to prove the reduction ___1___. Assume that ___2___ is decidable. The function f that maps instances of ___3___ to instances of ___4___ is performed by TM F given by the following pseudocode.

$F =$ "On input $\langle$___5___$\rangle$:

1. Construct the following M' given by the following pseudocode.
   $M' =$ "___6___"
2. Output $\langle$___7___$\rangle$"

Now, $\langle$___7___$\rangle$ is an element of ___8___ iff $\langle$___5___$\rangle$ is an element of ___9___. So using f and the assumption that ___2___ is decidable, we can decide ___10___. A contradiction. Therefore, ___2___ is undecidable. (This also means that the complement of ___2___ is undecidable; the complement of any undecidable language is itself undecidable.)

Figure 2: Proof template that may be appropriate for question 2a.

OLLSCOIL NA hÉIREANN MÁ NUAD
NATIONAL UNIVERSITY OF IRELAND MAYNOOTH

M.Sc. in Software Engineering Examination

SEMESTER 1
2005-2006

MATHEMATICS AND THEORY OF COMPUTER SCIENCE
PAPER CS605

Dr. Ita Richardson, Prof. Ronan Reilly, Mr. Tom Naughton

Declaration

To be signed by the student and collected by an invigilator at the beginning of the examination

1. I have searched through my copies of M. Sipser, *Introduction to the Theory of Computation*, first and second editions, (the Sipser book) and it does not contain any extra pages or annotations (except for annotations that correct minor typographical errors).
2. I understand that by failing to notify an invigilator of any annotations or extra pages in my copies of the Sipser book, I will receive a mark of zero in this examination. This does not affect any further disciplinary actions that the University may wish to take.
3. I understand also that directly copying large amounts of material from the Sipser books without substantially tailoring it to the question asked may result in a mark of zero.

Print name ________________________ Student number ______________

Signed ________________________ Date ______________