Be sure that this examination has 10 pages including this cover

The University of British Columbia

Final Examinations - December 2010

Computer Science 421/501

Closed book examination | Time: 150 minutes

Name ____________________ **Signature** ____________________

Student Number ____________________ **Instructor's Name** ____________________

Section Number ____________________

Special Instructions:

Calculators, notes, or other aids may not be used. The course blog ("very sketchy notes") will be provided. Answer questions on the exam. This exam is two-sided!

Rules governing examinations

1. Each candidate should be prepared to produce his library/AMS card upon request.

2. Read and observe the following rules:

No candidate shall be permitted to enter the examination room after the expiration of one half hour, or to leave during the first half hour of the examination.

Candidates are not permitted to ask questions of the invigilators, except in cases of supposed errors or ambiguities in examination questions.

CAUTION - Candidates guilty of any of the following or similar practices shall be immediately dismissed from the examination and shall be liable to disciplinary action.

(a) Making use of any books, papers or memoranda, other than those authorized by the examiners.

(b) Speaking or communicating with other candidates.

(c) Purposely exposing written papers to the view of other candidates. The plea of accident or forgetfulness shall not be received.

3. Smoking is not permitted during examinations.

1		10
2		10
3		10
4		10
5		10
6		10
7		10
8		10
Total		80

Marks

[10] **1.** Let L be the language of strings, w, over $\Sigma = \{0, 1\}$ such that the number of ones in w an odd number. Show that L is regular by exhibiting a DFA for L, and explain why your DFA accepts L.

Continued on page 3

[10] **2.** Let L the language of strings, w, over $\Sigma = \{0, 1\}$ such that the number of ones in w equal the number of zeros in w. Show that L is not regular by using the Myhill-Nerode theorem.

Continued on page 4

[10] **3.** Let $L = \{0^n 1^n 2^n \mid n = 0, 1, 2, \dots\}$. Use the pumping lemma for CFG's to show that L is not context-free.

Continued on page 5

[10] **4.** Let $L = \{0^m 1^m \mid m = 0, 1, 2, \dots\}$. Describe a 1-tape Turing machine that decides L in polynomial time. You should **explicitly write** and **explain** each of $Q, \Gamma, q_0, q_{\text{accept}}, q_{\text{reject}}, \delta$. You should justify that your algorithm takes at most polynomial time.

Continued on page 6

[10] **5.**

(a) Explain how to reduce 3SAT to SUBSET-SUM (by a polynomial time reduction). (Recall that SUBSET-SUM is the language of sequences of integers such that the last one is a sum of some subcollection of the others.)

(b) State what is means for a language to be NP-complete. Given that 3SAT is NP-complete, use part (a) to show that SUBSET-SUM is NP-complete.

Continued on page 7

[10] **6.** Let f be the function on positive integers given by $f(n) = 3n+1$ if n is odd, and $f(n) = n/2$ if n is even. Let L be the language of strings $x \in \{0, 1, \ldots, 9\}^*$ such that, viewing x as a base ten integer, all iterates of f on x (i.e., $x, f(x), f(f(x)), f(f(f(x))), \ldots$) are at most x^{10}. (For example, the iterates of 3 are $3, 10, 5, 16, 8, 4, 2, 1, 4, 2, 1, \ldots$, and since $16 \leq 3^{10}$ we have that $3 \in L$.) Show that L is in PSPACE.

Continued on page 8

[10] **7.** In two or three paragraphs, explain why in the course we studied (un)decidable and (un)acceptable (or (un)recognizable) languages. You should touch on the following questions: What did we learn just from counting considerations (e.g., if "$|\mathcal{P}| < |2^{\mathcal{I}}|$")? What did we learn about L_{yes} and the halting problem? What are undecidable or unacceptable problems that you might encounter in practice?

Continued on page 9

[10] **8.** In two or three paragraphs, outline how to prove the pumping lemmas for regular languages and for context-free languages and the Myhill-Nerode theorem. What idea(s) is common to the proofs of these theorems? Is there hope of extending this idea(s) to Turing machine computations (e.g., producing undeciable languages, or languages not in P)? Explain.

Continued on page 10

The End