Linköpings Tekniska Högskola
Institutionen för Datavetenskap
Gustav Nordh

2012-10-24

TDDD65 Introduction to the Theory of Computation
Examination, Wednesday, 2012-10-24

Material allowed:	An English dictionary (to any language) is allowed. Other material (like books, lecture notes, own notes etc.) and electronic equipment (computers, calculators, mobile phones etc.) is **not allowed**.
Questions:	Gustav Nordh, 0739 871855, will appear in the examination room around 1.5 hours after the start of the exam
Grading	There are 7 problems giving max 24 points. To pass you need 11 points. The lower bounds of points for the grades 3,4,5 are as follows: 3:12, 4:16, 5:20.
Results	An announcement will be posted at the course homepage http://www.ida.liu.se/~TDDD65 approx. one week after the exam with information on where you can look at your graded exam and discuss the result with the examiner.

Please observe the following:

- Solutions to different problems should be placed one-sided on separate page(s).
- Justify your answers properly: missing or insufficient explanations will result in reduction of points.
- Be sure that your answers are readable.
- Leave space for comments.

Good luck!

1. The well known NP-complete 1-in-3-SAT problem is the problem of checking the satisfiability of formulas that are the conjunction of clauses of the form $1/3(x, y, z)$, where the clause $1/3(x, y, z)$ is satisfied if EXACTLY one of the three variables is true. For example, (5 p)

$$1/3(x_1, x_2, x_3) \wedge 1/3(x_1, x_2, x_4) \wedge 1/3(x_1, x_4, x_5)$$

is a satisfiable 1-in-3-SAT formula.

Let 1-in-3-SAT$\neq\neq\neq$ denote the problem of checking the satisfiability of formulas that are the conjunction of clauses of the form $1/3(x, y, z) \wedge (x \neq p) \wedge (y \neq q) \wedge (z \neq r)$, i.e., 1-in-3-SAT clauses with three extra variables that are the negations of the variables in the 1-in-3-SAT clause.

(a) Is (1)

$$1/3(x_1, x_2, x_3) \wedge 1/3(x_1, x_2, x_4) \wedge 1/3(x_1, x_3, x_4) \wedge 1/3(x_2, x_3, x_4)$$

a satisfiable 1-in-3-SAT formula? In case it is, give a satisfying assignment, otherwise, motivate why there is no satisfying assignment.

(b) Prove that 1-in-3-SAT$\neq\neq\neq$ is NP-complete by giving a reduction from 1-in-3-SAT. (4)

BONUS: The 1-in-3-SAT$\neq\neq\neq$ problem was recently shown to be the easiest NP-complete SAT problem, and hence it is perhaps a good starting point for attacking the P vs NP question. If you are able to show (within 6 months from this exam and before anyone else) that this problem is in P (or that it is not in P if that is the case) you will be given the highest score for this course, regardless of your result on the exam.

2. Consider the following CFG where a, b, c, d, e are terminals, S, A, B, C are variables, and S is the start variable: (4 p)
$S \rightarrow AS \mid BS \mid CC$
$A \rightarrow aa \mid b$
$B \rightarrow a \mid bb$
$C \rightarrow c \mid d \mid e$

(a) Show a left-most derivation for the string *abcd*. (2)

(b) Is the grammar ambiguous? (Do not forget to justify your answer.) (2)

3. Consider the language L (over the alphabet $\Sigma = \{0, 1\}$) consisting of all strings where no 1 can come before a 0 and that have (strictly) more 0's than 1's. For example, $00011 \in L$, but $00101 \notin L$, and $0011 \notin L$. (4 p)

Use the pumping lemma to prove that L is nonregular.

4. Consider the languages A, B, and C, where A is a context-free language, B is not Turing recognizable, and C is recognized by a Turing machine that loops on some inputs. Given the following mapping reductions: (3 p)

- $D \leq_m A$,
- $C \leq_m F$,
- $F \leq_m C$,
- $C \leq_m B$, and
- $B \leq_m E$,

what can you conclude about the decidability of the languages D, E, and F? More specifically for each of D, E, and F answer whether it is decidable, undecidable, or whether the information given is not enough to draw a conclusion.

5. Provide a regular expression for the language over $\{0, 1\}$ consisting of all strings that are the binary representation of odd natural numbers. (2 p)

6. The language L over the alphabet $\Sigma = \{0, 1, 2\}$ consists of all strings over Σ containing at least two 0's or at least two 1's. So for example, $010012 \in L$ and $20200 \in L$ but $02212 \notin L$. (4 p)

   (a) Draw the transition graph of a 4-state NFA N that recognize L. (2)

   (b) Transform N into an equivalent DFA using the subset construction. (2)

7. (2 p)

   (a) Explain why the P vs. NP problem is so important. (1)

   (b) Do you believe that P $\neq$ NP? Write a short argument (at most 100 words) supporting your point of view. (1)