MSc in Computer Science
MSc in Mathematics and the Foundations of Computer Science

Michaelmas Term 2017

Foundations of CS

Exercise class 6.

1. Recall that a graph is 3-colorable if there a function f from the nodes of the graph to a set of colors of size at most 3 such that no two adjacent nodes have the same color.

   Let $3COL = \{\langle G\rangle : G$ is a finite 3-colorable graph $\}$

   Note that we are considering any graph here, not just planar graphs.

   Describe a polynomial time reduction from $3COL$ to SAT.

   Show what your reduction does on the *Peterson graph* a 10 node graph consisting of a pentagon with a star inscribed within it. The inner star consists of 5 points, with each pointed connected to the point two away; "inscribed" means that each point on the inner star is connected to one node on the outer pentagon.

   See the picture here:

   `http://en.wikipedia.org/wiki/Graph_coloring`

   You do not have to write out your solution to the Peterson graph coloring problem completely (you can use "..."). However, for those of you who are curious, you can put the full formula into the WINSAT solver:

   `http://users.ecs.soton.ac.uk/mqq06r/winsat/#Download_WinSat`

   To use WINSAT, you would have to write out your propositional formula in CNF in the "DIMACS format", and then load it into WINSAT. The DIMACS format is described here:

   `http://we.logoptimize.it/index.php/the-input-format/dimacs`

2. Let M be a Turing Machine with a read-only input-tape and a read/write "work-tape": such a machine works like a two-tape machine described in lectures, but only the work-tape can be written.

   Suppose w is a string in the input language of M on which M halts. The *space usage* of M on w is the number of tape cells on the work tape that are written on during the halting run.

   For a function f from integers to integers, we say M is a $SPACE(f(n))$-machine if on every input w of size n, M runs on w using space in $O(f(n))$.

   We say that a language L is in the class $SPACE(f(n))$ if there is some $SPACE(f(n))$-machine that decides L.

   The class $PSPACE$ is defined in the obvious way as $\bigcup_k SPACE(n^k)$ – that is, the set of languages that are accepted by a machine using at most polynomial amount of space on the worktape. The class of languages $LOGSPACE$ is defined as $SPACE(log(n))$.

(a) Give an example of a language decidable in LOGSPACE.

(b) Show that $LOGSPACE \subseteq PTIME \subseteq PSPACE$. Explain briefly how your argument for the first containment generalizes to show $PSPACE \subseteq EXPTIME$.

(c) Show that the following problem is in PSPACE: given two NFAs A_1 and A_2, is $L(A_1) = L(A_2)$ (i.e. the *equivalence problem for NFAs*).

Hint: consider an algorithm that searches for a counterexample to equivalence.

3. In the following exercises you are given a vocabulary V, a sentence ϕ in the vocabulary, and a structure M. Tell whether or not the sentence is true in the structure, explaining your answer.

(a) $V = \{<, *\}$
$\phi = \forall y \; \exists x \; (x < y \wedge y < x * x)$
$Dom(M)$ = the positive integers with $<, *$ the usual inequality and multiplication.

(b) $V = \{<\}$
$\phi = \forall x \; (\exists y \; x < y)$ $Dom(M) = \{1 \ldots 5\}$, with $<$ the usual inequality relation (restricted to these numbers).

(c) $V = \{<, \geq\}$
$\phi = (\forall x \forall y \; (x < y \leftrightarrow \neg(y \geq x))) \wedge ((\forall x \; \exists y \; x < y) \vee (\exists x \; \forall y \; x \geq y))$
$Dom(M) = \{1 \ldots 5\}$, $<, \geq$ the usual inequality relations restricted to these numbers.

(d) $V = \{1, +\}$
$\phi = \forall y \; \exists x \; (x = y + 1 \wedge (\exists y \; (y = x + 1 \wedge \exists x \; (x = y + 1))))$
$Dom(M)$ = the positive integers, with $+$ the usual addition, 1 is 1

4. Is one of the sentences in the previous exercise a tautology? A contradiction?