Eidgenössische Technische Hochschule Zürich
Ecole polytechnique fédérale de Zurich
Politecnico federale di Zurigo
Swiss Federal Institute of Technology Zurich

Institute of Theoretical Computer Science
Emo Welzl, lecturer, and Timon Hertli, teaching assistant

Friday, April 11, 2014

Satisfiability of Boolean Formulas *Spring 2014*
Special Assignment Set 2

- The solution is due on **Friday, May 2, 2014, 10:15 (strict!)**. Please bring a print-out of your solution with you to the exercise session. If you cannot attend, you may alternatively send your solution as a PDF to `timon.hertli@inf.ethz.ch`. We will send out a confirmation that we have received your file. Make sure you receive this confirmation within the day of the due date, otherwise complain timely.
- Please solve the exercises carefully and then write a nice and complete exposition of your solution using a computer, where we strongly recommend to use LaTeX. A tutorial can be found at `http://www.cadmo.ethz.ch/education/thesis/latex`.
- For geometric drawings that can easily be integrated into LaTeX documents, we recommend the drawing editor IPE, retrievable at `http://ipe7.sourceforge.net/` in source code and as an executable for Windows.
- You are welcome to discuss the tasks with your colleagues, but we expect each of you to hand in your own, individual write-up. **You should not share your write-up or (even worse) your source code!**
- There will be three special assignments this semester. Each of them will be graded and the average grade will contribute 30% to your final grade.
- This is a theory course, which means: if an exercise does not explicitly say ”you do not need to prove your answer” or ”justify intuitively”, then a formal proof is **always** required.
- As with all exercises, the material covered in special assignments is relevant for the final exam.

Exercise 1 (Prefix-free Codes) *(50 points)*

Let Σ be an alphabet. For $x, y \in \Sigma^*$, we say x is a *prefix* of y if there is $z \in \Sigma^*$ such that $x \circ z = y$. Let $\mathcal{X} \subseteq \Sigma^*$. We call $\mathcal{X}$ a *prefix-free* if for no distinct $x_1, x_2 \in \mathcal{X}$, x_1 is a prefix of x_2.

(a) Let $\mathcal{X} \subseteq \Sigma^*$ be prefix-free and finite. Show that

$$\sum_{x \in \mathcal{X}} |\Sigma|^{-|x|} \leq 1.$$

Let $\mathcal{X} \subseteq \Sigma^*$ be a finite set, and let X be a random variable taking values in $\mathcal{X}$. Let $p(x) := \Pr(X = x)$. You can assume $p(x) > 0$ that for all $x \in \mathcal{X}$. The *binary entropy* of X is defined as

$$H(X) := E[-\log p(X)]$$

where log is the binary logarithm.

To prepare yourself for the next subtasks, please read the Proof of Lemma 6.1 (page 131) and familiarize yourself with Jensen's inequality (Appendix, page v). Observe how Jensen's inequality works with the concave function log and with the convex function $-\log$. Also remember linearity of expectation and $\log \frac{p}{q} = \log p - \log q$.

(b) Show that if $\mathcal{X}$ is prefix-free and X is chosen uniformly at random from $\mathcal{X}$ then $E[|X|] \geq \frac{\log |\mathcal{X}|}{\log |\Sigma|}$.

(c) Show that $H(X) \leq \log |\mathcal{X}|$.

(d) Show that if $\mathcal{X}$ is prefix-free then $E\left[\log \frac{p(X)}{|\Sigma|^{-|X|}}\right] \geq 0$.

(e) Show that if $\mathcal{X}$ is prefix-free then $E[|X|] \geq \frac{H(X)}{\log |\Sigma|}$.

Exercise 2 (More SAT Algorithms) *(50 points)*

Let F be a (≤ 3)-CNF formula F on n variables. Denote by $I(F)$ the size of a maximum independent set of 3-clauses of F (note that maximum means that we take the maximum over all possible independent sets). In the lecture, you have seen that 3-SAT can be decided in time $O(7^{I(F)} \cdot \text{poly}(n))$.

The goal of this exercise is to construct an algorithm deciding 3-SAT in time $O(6^{I(F)} \cdot \text{poly}(n))$ and then to apply the idea to k-SAT.

(a) Let F be a (≤ 3)-CNF formula with at least one 3-clause. Show how to obtain from F in polynomial time (≤ 3)-CNF formulas F_j, $j \in \{1, 2, \ldots, 6\}$ such that

   (i) F is satisfiable if and only for some j, F_j is satisfiable.

   (ii) For all j, $I(F) \geq I(F_j) + 1$.

   Hint: First consider the case when there are 3-clauses conflicting in at least one variable.

   For the general case, pick a variable x and show that you can either "reach" 3-clauses from x and $\bar{x}$, or that you can "simplify" the formula (and then repeat the argument).

(b) Give an algorithm deciding 3-SAT in time $O(6^{I(F)} \cdot \text{poly}(n))$.

(c) Give an algorithm deciding k-SAT (for $k \geq 3$) in time $O((2^k - 2)^{\frac{n}{k}} \cdot \text{poly}(n))$.