Exercises for the Lecture Practical SAT Solving | Summer Semester 2018 | Assignment 3

Institute for Theoretical Informatics, Karlsruhe Institute of Technology (KIT) 17.05.2018

Tomas Balyo, Carsten Sinz Due 31.05.2018

Exercise 1 (Resolution Proof) [3 points]

Construct a resolution proof of unsatisfiability for the following formula

$(x_3 \vee x_4 \vee \bar{x}_1 \vee x_5) \wedge (\bar{x}_3 \vee x_4 \vee x_5) \wedge (x_3 \vee \bar{x}_4 \vee \bar{x}_1) \wedge (x_1 \vee x_2) \wedge (x_1 \vee \bar{x}_2) \wedge (\bar{x}_1 \vee \bar{x}_5) \wedge (\bar{x}_3 \vee \bar{x}_4 \vee x_5)$

Exercise 2 (Hidden Horn ⊆ SLUR) [5 points]

Prove that every hidden Horn formula is a SLUR formula, i.e., that the SLUR algorithm from slide 20 of Lecture 3 would never give up if its input is a hidden Horn formula.

Exercise 3 (Tseitin Encoding) [4 points]

Encode the following formula into CNF using the Tseitin Encoding.

$(\overline{x_1} \wedge \overline{(x_3 \iff x_2)}) \vee ((x_3 \implies \overline{x_4}) \wedge (x_1 \implies (x_2 \wedge \overline{x_3})) \wedge (x_4))$

Exercise 4 (DPLL) [5 points]

Simulate modern DPLL (from Slide 23 of Lecture 4 slides) by hand on the formula below. Select branching literals in the order $x_1, x_2, x_3, \ldots$.

$(x_3 \vee x_4 \vee \bar{x}_1 \vee x_5) \wedge (\bar{x}_3 \vee x_4 \vee x_5) \wedge (x_3 \vee \bar{x}_4 \vee \bar{x}_1) \wedge (x_1 \vee x_2) \wedge (x_1 \vee \bar{x}_2) \wedge (\bar{x}_1 \vee \bar{x}_5) \wedge (\bar{x}_3 \vee \bar{x}_4 \vee x_5)$

Exercise 5 (Local Search Challenge) [10(+10) points]

Implement a (stochastic) local search SAT solver. Follow the SAT Competition input/output format `http://www.satcompetition.org/2004/format-solvers2004.html` For a working solver you get 10 points. The author of the best solver receives a bonus of 10 points. The solvers will be evaluated on satisfiable random 3-SAT problems. (like the ones here: `https://baldur.iti.kit.edu/sat/files/local-sat.zip`). You don't need to start from scratch, use solver stub from the `local-sat.zip` package, it already contains the input parsing