SAARLAND UNIVERSITY ARTIFICIAL INTELLIGENCE
Prof. Dr. Jörg Hoffmann, Dr. Daniel Fišer, Daniel Höller, Dr. Sophia Saller

Practical Exercise Sheet 2

Deadline Friday, June 10, 23:59

About the submission of this sheet.

- You might submit the solutions to exercises in groups of up to 3 students.
- All students of a group need to be in the same tutorial.
- Hand in the solution **in CMS** and using "Team Groupings".
- Solutions need to be packaged into a `.zip` file that contains a single folder with name: `AI2022_PE2_mat1_mat2_mat3`, where `mat1, mat2, mat3` are the matriculation numbers of the students who submit together.
- This folder must contain the following files:
  - `authors.txt` listing names and matriculation numbers of all students of your group. Use one line per student and no spaces: Name;Matriculation number.
  - The Z3 files containing your solutions (see below).
- Do not add any other folder or sub folder, this means place all files directly into `AI2022_PE2_mat1_mat2_mat3`. Do not place any file outside of this folder.

1. (5×2 points) Let P, Q, R be atomic propositions. For each of the following formulas:

$$(P \to Q) \vee (Q \to P) \tag{1a}$$

$$R \to ((R \vee P) \wedge \neg P) \tag{1b}$$

$$(P \to R) \wedge (Q \to R) \wedge \neg(\neg Q \wedge (\neg R \vee P)) \tag{1c}$$

$$((P \vee Q) \leftrightarrow R) \vee (\neg R \vee \neg P) \tag{1d}$$

$$((P \wedge Q) \wedge (Q \to \neg R) \wedge (P \to R) \wedge (\neg Q \to R)) \tag{1e}$$

Use Z3 to find out whether the formula is satisfiable, unsatisfiable or a tautology. At the end of this sheet, you find a table summarizing the language fragment necessary to solve the given tasks. Do not use commands that are not given in that table.

For each $X \in \{$ 1a, 1b, 1c, 1d, 1e $\}$, submit your solution in the following way:

1. If X is a tautology, submit (place in your `.zip` file) the file `q`X`-tautology.z3` (e.g., `q1a-tautology.z3`, `q1b-tautology.z3`, ...) containing the Z3 program proving that X is a tautology.
2. If X is not a tautology, but it is satisfiable, then submit the file `q`X`-sat.z3` (e.g., `q1a-sat.z3`) containing a Z3 program that finds and prints out an assignment to P, Q, and R that satisfies the formula.
3. If X is not satisfiable, then submit the file `q`X`-unsat.z3` containing the Z3 program proving that X is unsatisfiable.

For each $X \in \{$ 1a, 1b, 1c, 1d, 1e $\}$, you can submit only one `.z3` file and you are **not** allowed to use truth tables in your solutions.

2. (20 points) Gregor, Natasha, Bob, Lise, Frank, Seifred, and Ivan meet at a pub and they want to order a drink. However, their order must satisfy the following conditions:

    1. Everybody must order either beer, or wine.
    2. Seifred will definitely order a wine.
    3. Bob orders a beer if and only if Frank orders a beer.
    4. If both Lise and Bob order a beer, then Natasha also orders a beer.
    5. Either Gregor and Lise both order a beer, or they both order a wine.
    6. If Frank orders a wine, then Gregor does not order a beer.
    7. At least one of Ivan, Gregor, and Lise orders a beer.
    8. Lise orders a beer if and only if Seifred orders a wine.

    Write a program in Z3 to find out who orders what. Again, only use commands given in Table 1. Think carefully about how to formulate the problem in propositional logic.

    Submit your solution in the file `q2.z3`, and describe in a few sentences how you decided to solve this problem and what solution you did find. The description (and the solution) must be placed as a comment at the beginning of the submitted file.

Statement	Description
General	
`(check-sat)`	Checks whether the model defined up to this point is satisfiable.
`(declare-const X Bool)`	Declares a new proposition with name `X`.
`(assert E)`	Adds boolean expression `E` as constraint.
`(get-value (E))`	Prints the value of `E`, where `E` can be an arbitrary expression such as constant, variable, function, or mathematical or boolean combination thereof (must occur after `(check-sat)`).
`(get-model)`	Prints all variable assignments (must occur after `(check-sat)`).
`(echo "message")`	Prints `message` to the console.
`; This is a comment`	Commenting.
Boolean Expressions	
`true`	Constant for true.
`false`	Constant for false.
`(not` E`)`	Negation of the boolean expression E.
`(and` E_1 `...` E_n`)`	Conjunction over the boolean expressions E_1 to E_n.
`(or` E_1 `...` E_n`)`	Disjunction over the boolean expressions E_1 to E_n.
`(implies` E_1 E_2`)`	Implication $E_1 \rightarrow E_2$.

Table 1: Z3 input language fragment you may use for this exercise.