

## Logic – Limboole

### Information about limboole (Checking Satisfiability and Validity of prop. formulas)

We use the tool `limboole` to check the satisfiability or validity of propositional formulas. This tool has been developed at the University of Linz by the team of Armin Biere. Their homepage can be found at:

<http://fmv.jku.at/limboole/>

The tool is available from their webpage and can be installed under Linux and Windows. For Linux the program is only available as source code. Hence we suggest to use the “`limboole.tar.gz`” provided on the homepage of this lecture, as it contains all dependencies and a simple `build.sh` script, which just needs to be executed to compile and link the program.

The formulas you want `limboole` to check are encoded as:

comment	starting with % until the end of the line
variable	string consisting of alphanumeric characters and - _ . [ ] \$ @
$\neg$	!
$\wedge$	&
$\vee$	
$\rightarrow$	->
$\leftrightarrow$	<->

Expressions may be put into parentheses and the precedences of the junctors are the same as presented in the lecture. As an example the formula  $(A_1 \leftrightarrow A_2) \wedge (A_2 \vee \neg A_3)$  is given to `limboole` as `(A1 <-> A2) & (A2 | !A3)`.

`Limboole` expects the formula it is going to analyse to be contained in a text file:

```
limboole <TEXT_FILE>
```

On Linux it is also possible to give the (short) formula as

```
echo "<FORMULA>" | ./limboole
```

`Limboole` then has two different modes of operation:

**Validity** This is the default mode if no further arguments are passed to `limboole`. The program then either outputs

```
% VALID formula
```

if the analysed formula is valid, or

```
% INVALID formula (falsifying assignment follows)
```

otherwise, together with an assignment which is not a model of the formula.

**Satisfiability** This mode is enabled by passing `-s` to `limboole`, i.e.

```
limboole -s <TEXT_FILE>
```

In this mode, the program tries to find a satisfying assignment of the formula. This is then output with

```
    % SATISFIABLE formula (satisfying assignment follows)
if it succeeds. Else it outputs
    % UNSATISFIABLE formula
```