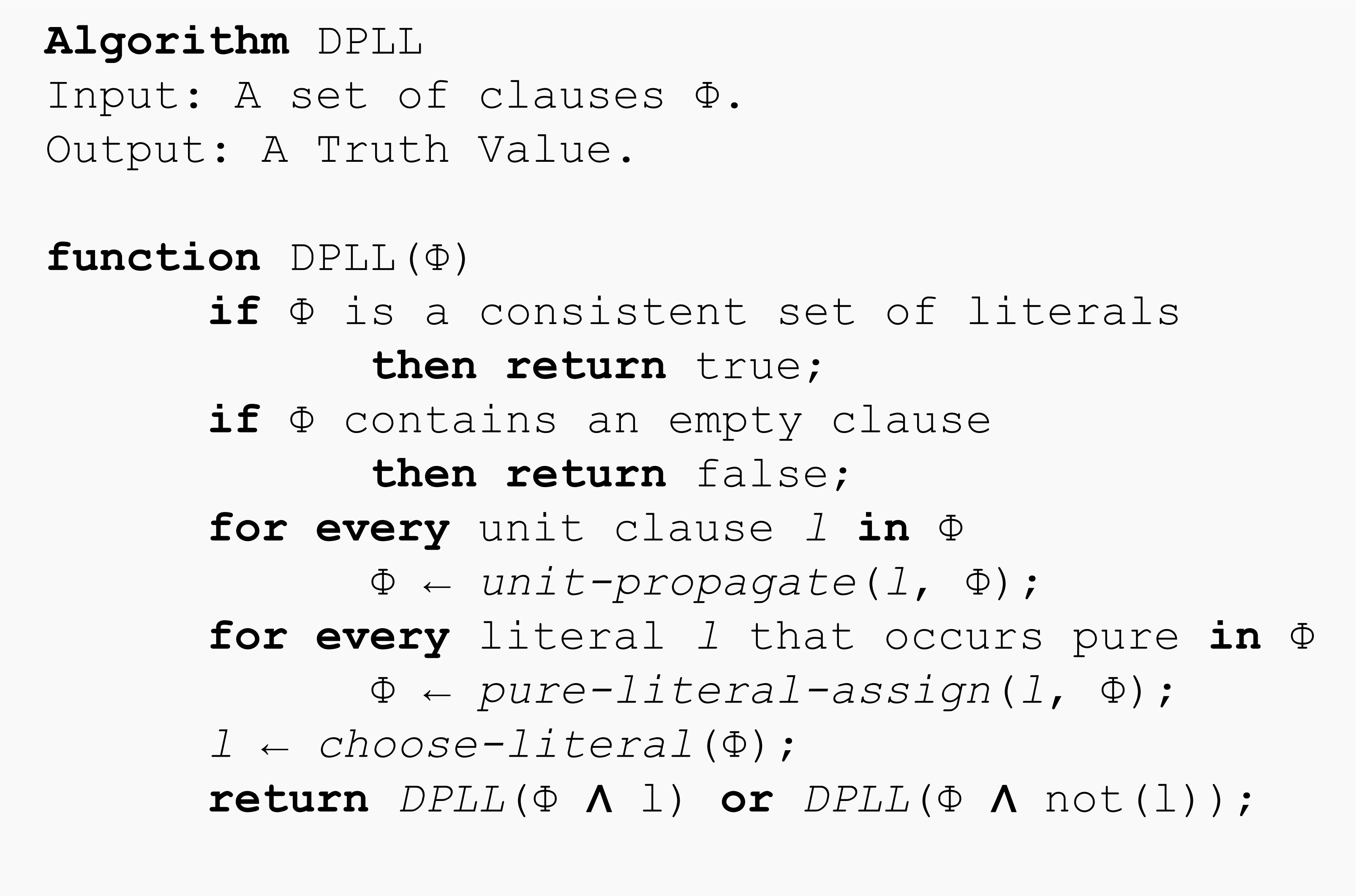
Davis–Putnam–Logemann–Loveland (DPLL) algorithm

DPLL is a complete, backtracking-based search algorithm for deciding the satisfiability of propositional logic formulae in conjunctive normal form, i.e. for solving the CNF-SAT problem.



```
Algorithm DPLL
Input: A set of clauses Φ.
Output: A Truth Value.

function DPLL(Φ)
      if Φ is a consistent set of literals
            then return true;
      if Φ contains an empty clause
            then return false;
      for every unit clause l in Φ
            Φ ← unit-propagate(l, Φ);
      for every literal l that occurs pure in Φ
            Φ ← pure-literal-assign(l, Φ);
      l ← choose-literal(Φ);
      return DPLL(Φ ∧ l) or DPLL(Φ ∧ not(l));
```

Davis–Putnam–Logemann-Loveland (DPLL) algorithm (cont.)

Where:

- `unit-propagate(l, Φ)` and `pure-literal-assign(l, Φ)` are functions that return the result of applying unit propagation and the pure literal rule, respectively, to the literal `l` and the formula `Φ`.

  They replace every occurrence of `l` with "true" and every occurrence of `not` with "false" in the formula `Φ`, and simplify the resulting formula.

- `Φ ∧ l` denotes the simplified result of substituting "true" for `l` in `Φ`.