Questionnaire "Logic and Computability"

Summer Term 2024

Contents

5 Combinational Equivalence Checking

5.1 Normal Forms

5.1.1 Define the *Disjunctive Normal Form (DNF)* of formulas in propositional logic. Use the proper terminology and give an example.

Solution

There is no solution available for this question yet.

5.1.2 Define the *Conjunctive Normal Form (CNF)* of formulas in propositional logic. Use the proper terminology and give an example.

Solution

There is no solution available for this question yet.

5.1.3 Given a formula in propositional logic. Explain how to extract a *CNF* representation as well as a *DNF* representation of φ using the truth table from φ.

5.1.4 Given the formula $\varphi = (q \rightarrow p) \wedge (r \vee \neg p)$. Compute its representation in Disjunctive Normal Form (*DNF*) using a truth table.

5.1.5 Given the formula $\varphi = (q \rightarrow p) \wedge (r \vee \neg p)$. Compute its representation in Conjunctive Normal Form (*CNF*) using a truth table.

5.1.6 Given the formula $\varphi = (a \wedge \neg b \wedge \neg c) \vee ((\neg c \rightarrow a) \rightarrow b)$. Use the truth table of φ to compute its representation in (a) CNF and (b) DNF.

5.1.7 Given the formula $\varphi = (q \rightarrow \neg r) \wedge \neg(p \vee q \vee \neg r)$. Use the truth table of φ to compute its representation in (a) CNF and (b) DNF.

5.1.8 Given the formula $\varphi = \neg(a \rightarrow \neg b) \vee (\neg a \rightarrow c)$. Use the truth table of φ to compute its representation in (a) CNF and (b) DNF.

5.1.9 Consider the propositional formula $\varphi = (\neg(\neg a \wedge b) \wedge \neg c)$. Fill out the truth table for φ and its subformulas. Compute a CNF as well as a DNF for φ from the truth table.

a	b	c	$\neg a$	$\neg a \wedge b$	$\neg(\neg a \wedge b)$	$\neg c$	$\varphi = (\neg(\neg a \wedge b) \wedge \neg c)$
F	**F**	**F**					
F	**F**	**T**					
F	**T**	**F**					
F	**T**	**T**					
T	**F**	**F**					
T	**F**	**T**					
T	**T**	**F**					
T	**T**	**T**					

5.1.10 Consider the propositional formula $\varphi = (p \vee \neg q) \rightarrow (\neg p \wedge \neg r)$. Fill out the truth table for φ and its subformulas. Compute a CNF as well as a DNF for φ from the truth table.

p	q	r	$\neg q$	$p \vee \neg q$	$\neg p$	$\neg r$	$\neg p \wedge \neg r$	$\varphi = (p \vee \neg q) \rightarrow (\neg p \wedge \neg r)$
F	**F**	**F**	**T**	**T**	**T**	**T**	**T**	**T**
F	**F**	**T**	**T**	**T**	**T**	**F**	**F**	**F**
F	**T**	**F**	**F**	**F**	**T**	**T**	**T**	**T**
F	**T**	**T**	**F**	**F**	**T**	**F**	**F**	**T**
T	**F**	**F**	**T**	**T**	**F**	**T**	**F**	**F**
T	**F**	**T**	**T**	**T**	**F**	**F**	**F**	**F**
T	**T**	**F**	**F**	**T**	**F**	**T**	**F**	**F**
T	**T**	**T**	**F**	**T**	**F**	**F**	**F**	**F**

The resulting CNF:

$$
\begin{aligned}
(p \vee q \vee \neg r) &\wedge \\
(\neg p \vee q \vee r) &\wedge \\
(\neg p \vee q \vee \neg r) &\wedge \\
(\neg p \vee \neg q \vee r) &\wedge \\
(\neg p \vee \neg q \vee \neg r)&
\end{aligned}
$$

The resulting DNF:

$$
\begin{aligned}
(\neg p \wedge \neg q \wedge \neg r) &\vee \\
(\neg p \wedge q \wedge \neg r) &\vee \\
(\neg p \wedge q \wedge r)&
\end{aligned}
$$

5.1.11 Look a the following statements and tick all items that conform to a *DNF*.

- ☐ $a \vee b$
- ☐ A DNF is a conjunction of clauses.
- ☐ $(a \vee b) \wedge (\neg b \vee \neg a \vee c) \wedge \neg c$
- ☐ $(a \wedge b) \vee (\neg b \wedge \neg a \wedge c) \vee \neg c$
- ☐ A DNF is a conjunction of disjunctions of literals.
- ☐ b
- ☐ $a \wedge b \wedge \neg c$
- ☐ $(\neg a \wedge b) \wedge (\neg a \wedge c)$
- ☐ A DNF is a disjunction of cubes.
- ☐ $\neg(a \wedge \neg b) \wedge c$
- ☐ A DNF is a disjunction of conjunctions of literals.
- ☐ $a \wedge \neg b$

5.1.12 In the following list, tick all items that conform to the Conjunctive Normal Form (CNF).

- ☐ $(a \wedge b \wedge \neg c) \vee (\neg b \wedge \neg c) \vee (e \wedge \neg f)$
- ☐ a
- ☐ $\neg b$
- ☐ $a \wedge \neg b$
- ☐ $a \vee \neg b$
- ☐ $a \vee (\neg b \wedge c)$
- ☐ $(a \vee \neg b) \wedge c$
- ☐ $\neg(p \vee q)$
- ☐ $x \vee \neg y \vee z$

5.1.13 In the following list, tick all items that conform to the Disjunctive Normal Form (DNF).

- ☐ $(a \wedge b \wedge \neg c) \vee (\neg b \wedge \neg c) \vee (e \wedge \neg f)$
- ☐ $(a \vee b \vee \neg c) \wedge (\neg b \vee \neg c) \wedge (e \vee \neg f)$
- ☐ $\neg b$
- ☐ $a \wedge \neg b$
- ☐ $a \vee \neg b$
- ☐ $a \vee (\neg b \wedge c)$
- ☐ $(a \vee \neg b) \wedge c$
- ☐ $\neg(p \vee q)$
- ☐ $x \vee \neg y \vee z$

5.2 Relations between Satisfiability, Validity, Equivalence and Entailment

5.2.1 Explain the duality of *satisfiability* and *validity*.

Solution

There is no solution available for this question yet.

5.2.2 How can you check whether it is true that $\varphi \models \psi$ using a decision procedure for (a) *satisfiability* or (b) *validity*?

Solution

There is no solution available for this question yet.

5.2.3 *A formula φ is valid, if and only if $\neg\varphi$ is not satisfiable.* Explain why this statement holds true.

Solution

There is no solution available for this question yet.

5.2.4 Given two propositional logic formulas φ and ψ. How can we check whether $\varphi \equiv \psi$ using a decision procedure for (a) satisfiability, (b) for validity, and (c) for semantic entailment?

Solution

There is no solution available for this question yet.

5.2.5 Given a propositional logic formula φ. How can we check whether φ is *valid* using a decision procedure for (a) satisfiability and (b) equivalence?

Solution

There is no solution available for this question yet.

5.2.6 Given a propositional logic formula φ. Tick all statements that are true.

- ☐ A formula φ is *valid*, if and only if $\neg\varphi$ is *satisfiable*.
- ☐ A formula ψ is *satisfiable*, if and only if $\neg\varphi$ is *valid*.
- ☐ A formula φ is *satisfiable*, if and only if $\neg\varphi$ is *not valid*.
- ☐ A formula φ is *valid*, if and only if $\neg\varphi$ is *not satisfiable*.

5.2.7 Given two propositional logic formulas φ and ψ. Tick all statements that are true.

- ☐ If $\neg\varphi$ is not satisfiable, φ is not valid.
- ☐ If $\top \models \varphi$, φ is valid.
- ☐ If $\varphi \leftrightarrow \psi$ is valid, φ entails ψ.
- ☐ If $\varphi \rightarrow \psi$ is valid, both formulas are equivalent.

5.2.8 Given two propositional logic formulas φ and ψ. Tick all statements that are true.

- ☐ If $\varphi \land \neg\psi$ is not satisfiable, φ entails ψ.

☐ If $\neg\varphi$ is not valid, φ is satisfiable.

☐ If φ entails ψ and ψ entails φ, both formulas are equivalent.

☐ If φ is equivalent to $\top$, φ is valid..

Solution

There is no solution available for this question yet.

5.3 Combinational Equivalence Checking

5.3.1 Explain the algorithm used to decide the equivalence of combinational circuits via the reduction to satisfiability.

Solution

Let C_1 and C_2 denote the two combinational circuits. In order to check whether C_1 and C_2 are equivalent, one has to perform the following steps:

(a) Encode C_1 and C_2 into two propositional formulas φ_1 and φ_2.

(b) Compute the Conjunctive Normal Form (CNF) of $\varphi_1 \oplus \varphi_2$, using Tseitin encoding; i.e., $CNF(\varphi_1 \oplus \varphi_2)$.

(c) Give the formula $CNF(\varphi_1 \oplus \varphi_2)$ to a SAT solver and check for satisfiability.

(d) C_1 and C_2 are equivalent if and only if $CNF(\varphi_1 \oplus \varphi_2)$ is UNSAT.

5.3.2 Give the definition of equisatisfiability.

Solution

Two propositional formulas φ and ψ are *equisatisfiable* if and only if either *both are satisfiable* or *both are unsatisfiable.*
When checking whether two formulas φ_1 and φ_2 are equivalent we check whether $\varphi = \varphi_1 \oplus \varphi_2$ is satisfiable. If φ is SAT we know that there is a model such that one of the input formulas evaluated to true, while the other evaluated to false. The equisatisfiable formula $CNF(\varphi)$ is satisfiable if and only if φ is satisfiable and therefore answers our question of whether the two input formulas are equivalent.

5.3.3 Given a propositional logic formula φ, the Tseitin transformation computes an equisatisfiable formula φ' in CNF. Why is this enough for equivalence checking?

Solution

There is no solution available for this question yet.

5.3.4 (a) What does it mean that two formulas φ and ψ are *equisatisfiable*? (b) Explain the difference between *satisfiability* and *equisatisfiability*.

Solution

There is no solution available for this question yet.

5.3.5 Explain the algorithm of *Tseitin transformation* to obtain an equisatisfiable formula in CNF. Give step-by-step instructions of how to apply Tseitin transformation to a propositional formula.
(Note: Focus on the concept. You do *not* need to quote the rewrite rules!)

Solution

There is no solution available for this question yet.

5.3.6 What is the advantage of applying *Tseitin transformation* to obtain an equisatisfiable CNF, especially compared to using truth tables?

Solution

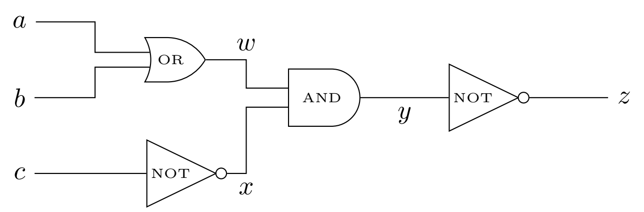
Given an original formula φ. The equisatisfiable formula in CNF after Tseitin encoding – $CNF(\varphi)$ – is *linear* in the size of φ, since the number of variables and clauses introduced by Tseitin encoding is *linear* in the size of φ. Using a truth table could result in an exponential blowup when constructing a CNF.

5.3.7 Derive a Rewrite-Rule for an implication node, i.e., what clauses are introduced by the node $x \leftrightarrow (p \rightarrow q)$?

Solution

$$
\begin{aligned}
x \leftrightarrow (p \rightarrow q) &\Leftrightarrow x \leftrightarrow (p \rightarrow q) \\
&\Leftrightarrow (x \rightarrow (p \rightarrow q)) \land ((p \rightarrow q) \rightarrow x) \\
&\Leftrightarrow (x \rightarrow (\neg p \lor q)) \land ((\neg p \lor q) \rightarrow x) \\
&\Leftrightarrow (\neg x \lor (\neg p \lor q)) \land (\neg(\neg p \lor q) \lor x) \\
&\Leftrightarrow (\neg x \lor \neg p \lor q) \land ((\neg\neg p \land \neg q) \lor x) \\
&\Leftrightarrow (\neg x \lor \neg p \lor q) \land ((p \land \neg q) \lor x) \\
&\Leftrightarrow (\neg x \lor \neg p \lor q) \land ((p \lor x) \land (\neg q \lor x)) \\
&\Leftrightarrow (\neg x \lor \neg p \lor q) \land (p \lor x) \land (\neg q \lor x)
\end{aligned}
$$

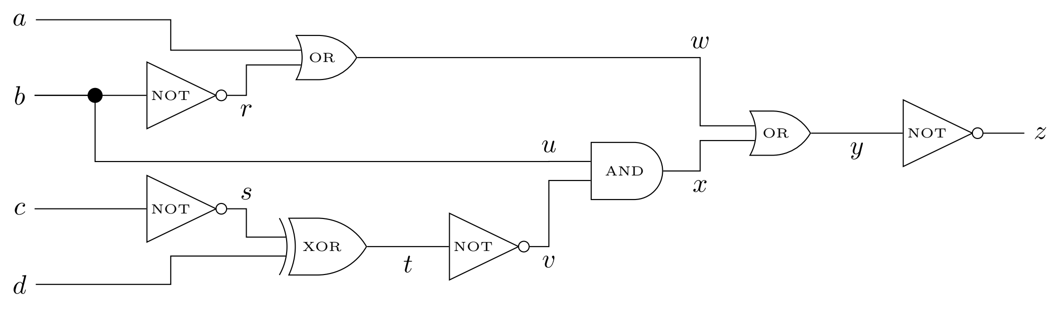
5.3.8 Compute the propositional formula φ represented by the following circuit. Furthermore, compute an equisatisfiable formula φ' using the Tseitin transformation.



Solution

There is no solution available for this question yet.

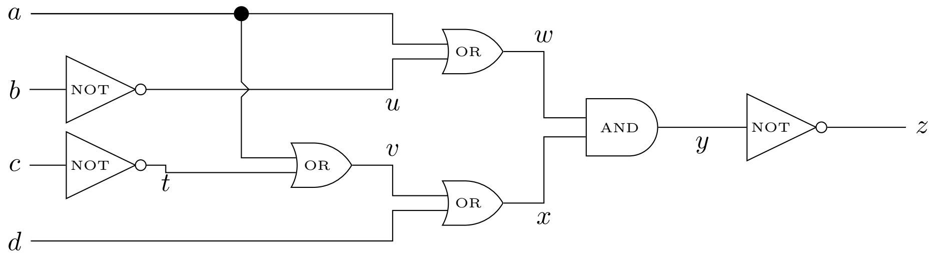
5.3.9 Compute the propositional formula φ represented by the following circuit. Furthermore, compute an equisatisfiable formula φ' using the Tseitin transformation.



Solution

There is no solution available for this question yet.

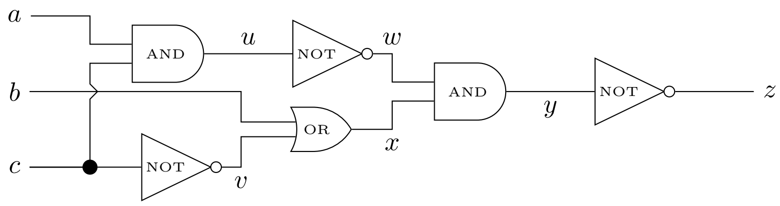
5.3.10 Compute the propositional formula φ represented by the following circuit. Furthermore, compute an equisatisfiable formula φ' using the Tseitin transformation.



Solution

There is no solution available for this question yet.

5.3.11 Compute the propositional formula φ represented by the following circuit. Furthermore, compute an equisatisfiable formula φ' using the Tseitin transformation.



Solution

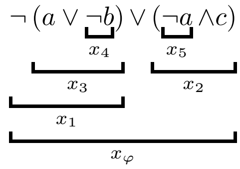
There is no solution available for this question yet.

We list the *Tseitin-rewriting rules* to be applied for the following examples.

$$\begin{aligned} \chi \leftrightarrow (\varphi \vee \psi) &\quad \Leftrightarrow (\neg\varphi \vee \chi) \wedge (\neg\psi \vee \chi) \wedge (\neg\chi \vee \varphi \vee \psi) \\ \chi \leftrightarrow (\varphi \wedge \psi) &\quad \Leftrightarrow (\neg\chi \vee \varphi) \wedge (\neg\chi \vee \psi) \wedge (\neg\varphi \vee \neg\psi \vee \chi) \\ \chi \leftrightarrow \neg\varphi &\quad \Leftrightarrow (\neg\chi \vee \neg\varphi) \wedge (\chi \vee \varphi) \end{aligned}$$

5.3.12 Apply the Tseitin transformation to $\varphi = \neg(a \vee \neg b) \vee (\neg a \wedge c)$. For each variable you introduce, clearly indicate which subformula it represents.

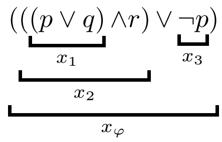
Solution



$$
\begin{aligned}
\varphi' = {} & x_\varphi \wedge \\
& (\neg x_1 \vee x_\varphi) \wedge (\neg x_2 \vee x_\varphi) \wedge (\neg x_\varphi \vee x_1 \vee x_2) \wedge \\
& (\neg x_1 \vee \neg x_3) \wedge (x_1 \vee x_3) \wedge \\
& (\neg a \vee x_3) \wedge (\neg x_4 \vee x_3) \wedge (\neg x_3 \vee a \vee x_4) \wedge \\
& (\neg x_2 \vee x_5) \wedge (\neg x_2 \vee c) \wedge (\neg x_5 \vee \neg c \vee x_2) \wedge \\
& (\neg x_4 \vee \neg b) \wedge (x_4 \vee b) \wedge \\
& (\neg x_5 \vee \neg a) \wedge (x_5 \vee a)
\end{aligned}
$$

5.3.13 Apply the Tseitin transformation to $\varphi = ((p \vee q) \wedge r) \vee \neg p$. For each variable you introduce, clearly indicate which subformula it represents.

Solution



$$
\begin{aligned}
\varphi' = {} & x_\varphi \wedge \\
& (\neg x_2 \vee x_\varphi) \wedge (\neg x_3 \vee x_\varphi) \wedge (\neg x_\varphi \vee x_2 \vee x_3) \wedge \\
& (\neg x_2 \vee x_1) \wedge (\neg x_2 \vee r) \wedge (\neg x_1 \vee \neg r \vee x_2) \wedge \\
& (\neg p \vee x_1) \wedge (\neg q \vee x_1) \wedge (\neg x_1 \vee p \vee q) \wedge \\
& (\neg x_3 \vee \neg p) \wedge (x_3 \vee p)
\end{aligned}
$$

5.3.14 Apply the Tseitin transformation to $\varphi = \neg(\neg b \wedge \neg c) \vee (\neg c \wedge a)$. For each variable you introduce, clearly indicate which subformula it represents.

Solution

There is no solution available for this question yet.

5.3.15 Apply the Tseitin transformation to $\varphi = (q \wedge \neg r) \vee \neg(q \wedge \neg r)$. For each variable you introduce, clearly indicate which subformula it represents.

Solution

There is no solution available for this question yet.

5.3.16 Apply the Tseitin transformation to $\varphi = (\neg(\neg a \wedge b) \wedge \neg c)$. For each variable you introduce, clearly indicate which subformula it represents.

Solution

There is no solution available for this question yet.

5.3.17 Apply the Tseitin transformation to $\varphi = (p \lor \neg q) \lor (\neg p \land \neg r)$. For each variable you introduce, clearly indicate which subformula it represents.

Solution

There is no solution available for this question yet.

5.3.18 Apply the Tseitin transformation to $\varphi = \neg(p \to q) \land (r \land p)$. For each variable you introduce, clearly indicate which subformula it represents. Derive the Tseitin transformation rule for $\to$ or transform the input such that you can use the rules above.

Solution

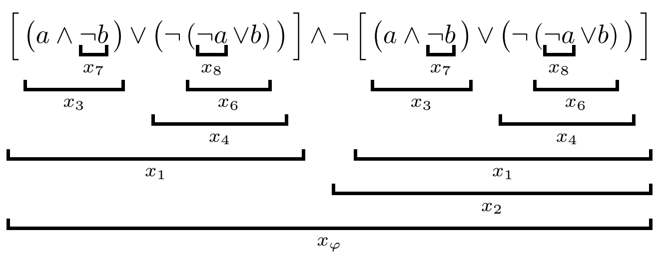
There is no solution available for this question yet.

5.3.19 Check whether $\varphi_1 = a \land \neg b$ and $\varphi_2 = \neg(\neg a \lor b)$ are semantically equivalent using the reduction to satisfiability. Follow the algorithm discussed in the lecture and state the final formula that is used as input for a SAT solver.

Solution

- We start by construction φ:

$$\begin{aligned}\varphi &= \varphi_1 \oplus \varphi_2 \\ &= [\varphi_1 \lor \varphi_2] \land \neg[\varphi_1 \land \varphi_2] = \\ &= [(a \land \neg b) \lor (\neg(\neg a \lor b))] \land \neg[(a \land \neg b) \land (\neg(\neg a \lor b))]\end{aligned}$$



$$\begin{aligned}\varphi' = x_\varphi \land & \\ &(\neg x_\varphi \lor x_1) \land (\neg x_\varphi \lor x_2) \land (\neg x_1 \lor \neg x_2 \lor x_\varphi) \land \\ &(\neg x_1 \lor \neg x_2) \land (x_1 \lor x_2) \land \\ &(\neg x_3 \lor x_1) \land (\neg x_4 \lor x_1) \land (\neg x_1 \lor x_3 \lor x_4) \land \\ &(\neg x_3 \lor a) \land (\neg x_3 \lor x_7) \land (\neg a \lor \neg x_7 \lor x_3) \land \\ &(\neg x_4 \lor \neg x_6) \land (x_4 \lor x_6) \land \\ &(\neg x_8 \lor x_6) \land (\neg b \lor x_6) \land (\neg x_6 \lor x_8 \lor b) \land \\ &(\neg x_7 \lor \neg b) \land (x_7 \lor b) \land \\ &(\neg x_8 \lor \neg a) \land (x_8 \lor a) \land\end{aligned}$$

5.3.20 Check whether $\varphi_1 = (a \wedge b) \vee \neg c$ and $\varphi_2 = (a \vee \neg c) \wedge (b \vee \neg c)$ are semantically equivalent using the reduction to satisfiability. Follow the algorithm discussed in the lecture and state the final formula that is used as input for a SAT solver.

Solution

There is no solution available for this question yet.