Prof. J. Esparza
M. Lazić
Balsubramanian A. R.

Exercise sheet: Propositional Logic[1]

Organisational notes.

Since this is the first exercise sheet its main focus is on aspects of propositional logic that we consider prerequisites of this course. Regardless, contrary to following exercise sheets we will discuss this sheet in two consecutive weeks rather than only one week to make sure these basics are well understood and to wait for more content of the lecture to ask questions about. Therefore, we discuss exercises (1) to (6) in the tutorial session on Wednesday 19th April, 2023 and exercises (7) to (13) on Wednesday 26th April, 2023.

Exercise 1: Facts and deductions

Let F, G and H be formulas and let $\mathcal{S}$ be a set of formulas. Which of the following statements are true? Justify your answer.

1. If F is unsatisfiable, then $\neg F$ is valid.
2. If $F \rightarrow G$ is satisfiable and F is satisfiable, then G is satisfiable.
3. $P_1 \rightarrow (P_2 \rightarrow (P_3 \rightarrow \ldots (P_n \rightarrow P_1)\ldots))$ is valid.
4. $\mathcal{S} \models F$ and $\mathcal{S} \models \neg F$ cannot both hold.
5. If $\mathcal{S} \models F \vee G$, $\mathcal{S} \cup \{F\} \models H$ and $\mathcal{S} \cup \{G\} \models H$, then $\mathcal{S} \models H$.

Exercise 2: Equivalence, equisatisfiability, equality

Let F and G be two formulas.

1. Explain the difference between F and G being **equisatisfiable** and them being **logically equivalent**.
2. Explain very briefly the difference between $F \leftrightarrow G$ and $F \equiv G$.

Exercise 3: Equational proof

Give an equational proof of the following equivalence, justifying each step with reference to the Boolean algebra axioms and the Substitution Rule as appropriate.

$$\neg((\neg P \vee Q) \wedge P) \vee Q \equiv \mathit{true}$$

[1] We gratefully acknowledge inspiration from Christoph Haase, the lecture Diskrete Strukturen, and various lectures of the "Lehr- und Forschungsgebiet für Mathematische Grundlagen der Informatik" at the RWTH Aachen.

Exercise 4: Deductions

Suppose that F and G are formulas such that $F \models G$.

1. Show that if F and G have no variable in common then either F is unsatisfiable or G is valid.

2. Now let F and G be arbitrary formulas with $F \models G$. Show that there is a formula H, mentioning only propositional variables common to F and G, such that $F \models H$ and $H \models G$.

   Hint. Recall that every truth table is realised by some propositional formula and consider what the truth table of H ought to look like: under which assignments must H be true and under which assignments must H be false?

Exercise 5: Perfect matching

A **perfect matching** in an undirected graph $G = (V, E)$ is a subset of the edges $M \subseteq E$ such that every vertex $v \in V$ is an endpoint of exactly one edge in M. Given a finite graph G, describe how to obtain a propositional formula F_G such that F_G is satisfiable if and only if G has a perfect matching. The formula F_G should be computable from G in time polynomial in $|V|$.

Exercise 6: U-assignment

Fix a non-empty set U. A U**-assignment** is a function from the collection of propositional variables to 2^U, the power set of U, that is, $\mathcal{A}$ maps each propositional variable to a subset of U. Such an assignment is extended to all formulas as follows:

- $\mathcal{A}(\mathit{false}) = \emptyset$ and $\mathcal{A}(\mathbf{true}) = U$;
- $\mathcal{A}(F \wedge G) = \mathcal{A}(F) \cap \mathcal{A}(G)$;
- $\mathcal{A}(F \vee G) = \mathcal{A}(F) \cup \mathcal{A}(G)$;
- $\mathcal{A}(\neg F) = U \setminus \mathcal{A}(F)$.

Say that a formula F is U**-valid** if $\mathcal{A}(F) = U$ for all U-assignments $\mathcal{A}$.

1. Show that if F is U-valid then F is valid with respect to the standard semantics defined in the lecture notes.

   Hint: Show that each standard assignment $\mathcal{A}$ can be "simulated" by a certain U-assignment $\mathcal{A}'$.

2. Show that if F is valid then F is U-valid.

   Hint: Fix an arbitrary $u \in U$ and argue that $u \in \mathcal{A}(F)$.

Exercise 7: Large disjunctive normal form

1. Write down a **DNF**-formula equivalent to $(P_1 \vee Q_1) \wedge (P_2 \vee Q_2) \wedge \cdots \wedge (P_n \vee Q_n)$.
2. Prove that any **DNF**-formula equivalent to the above formula must have at least 2^n clauses.

Exercise 8: Truth tables and syntax trees

Let p, q, r, s be propositional variables. We define

$$
\begin{array}{ll}
F_1 := \neg(p \wedge (\neg q \leftrightarrow p)) & F_2 := (\neg\neg\neg p \vee \neg\neg(q \vee (r \to p))) \\
F_3 := (((q \vee p) \leftrightarrow (q \wedge r)) \wedge \neg(p \to r)) & F_4 := ((q \leftrightarrow p) \wedge ((\neg p \to r) \wedge (\neg q \leftrightarrow r))) \\
F_5 := (((p \to r) \to (q \to r)) \to (p \to q)) & F_6 := ((p \oplus q) \to ((p \leftrightarrow s) \vee \neg r))
\end{array}
$$

1. Give for all formulae $F_1, \ldots, F_6$ a syntax tree.
2. Give for all formulae $F_1, \ldots, F_6$ a truth table.
3. Decide which formulae $F_1, \ldots, F_6$ are valid or unsatisfiable.
4. State which formulae $F_1, \ldots, F_6$ are equivalent.

Exercise 9: Truth tables and syntax trees

Let p, q, r, s be propositional variables. We define

$$
\begin{array}{l}
F_1 := ((((q \oplus q) \vee q) \wedge \neg(q \wedge p)) \wedge (((r \leftrightarrow p) \to (s \wedge q)) \oplus ((s \leftrightarrow s) \to (r \wedge s)))) \\
F_2 := (((\neg r \wedge (q \oplus p)) \to ((r \wedge s) \to (s \vee s))) \vee (((p \leftrightarrow r) \leftrightarrow \neg p) \to ((q \leftrightarrow p) \oplus (r \to p)))) \\
F_3 := (\neg((s \vee s) \leftrightarrow (p \leftrightarrow s)) \to r) \\
F_4 := \neg(((r \oplus p) \leftrightarrow \neg p) \to ((q \wedge q) \to r))
\end{array}
$$

1. Give for all formulae $F_1, \ldots, F_4$ a syntax tree.
2. Give for all formulae $F_1, \ldots, F_4$ a truth table.
3. Decide which formulae $F_1, \ldots, F_4$ are valid or unsatisfiable.
4. State which formulae $F_1, \ldots, F_4$ are equivalent.

Exercise 10: Modelling finite functions

Propositional logic can be used to model functions with finite domains. Therefore, assume there is a function $f\colon D \to V$ such that D and V are finite. W.l.o.g. we identify D with $\{0, \ldots, n\}$ and V with $\{0, \ldots, m\}$. We encode values from D by propositional values $x_0, \ldots, x_{\lceil \log_2(n) \rceil - 1}$ and values from V by propositional values $y_0, \ldots, y_{\lceil \log_2(m) \rceil - 1}$. Hence, we can identify f with a formula φ_f over the propositional values of $x_0, \ldots, x_{\lceil \log_2(n) \rceil - 1}$, $y_0, \ldots, y_{\lceil \log_2(m) \rceil - 1}$ such that

$$f(d) = v \text{ if and only if } \mathcal{I}_{d,v} \models \varphi_f,$$

where $\mathrm{bin}(k)_\ell$ is the ℓ-th bit of the binary representation of k and $\mathcal{I}_{d,v}$ is defined such that $\mathcal{I}_{d,v}(x_i) = \mathrm{bin}(d)_i$ and $\mathcal{I}_{d,v}(y_i) = \mathrm{bin}(v)_i$.

1. Give $\varphi_f, \varphi_g, \varphi_h$ for the functions

$$\begin{aligned}
f &\colon \{0,1,2,3\} \to \{0,1,2,3\} \text{ with } f(x) = (x+1) \bmod 4\\
g &\colon \{0,1,2,3\} \to \{0,1,2,3\} \text{ with } g(x) = (2x) \bmod 4\\
h &\colon \{0,1,2,3\} \to \{0,1,2,3\} \text{ with } h(x) = (x+3) \bmod 4
\end{aligned}$$

2. Prove your choice of $\varphi_f, \varphi_g, \varphi_h$.

Exercise 11: Modelling finite functions

1. Give $\varphi_f, \varphi_g, \varphi_h$ as above for

$$\begin{aligned}
&f\colon \{0,1,2,3,4,5,6,7\} \to \{0,1,2,3,4,5,6,7\} \text{ with } f(x) = (3x+1) \bmod 8\\
&g\colon \{0,1,2,3,4,5,6,7\} \to \{0,1,2,3,4,5,6,7\} \text{ with } g(x) = \min((3x+1) \bmod 8, x)\\
&h\colon \{0,1,2,3\} \times \{0,1,2,3\} \to \{0,1,2,3\} \text{ with } h(x,y) = \max(x,y)
\end{aligned}$$

2. Prove your choice of $\varphi_f, \varphi_g, \varphi_h$.

Exercise 12: Equivalence

Let A, B, C be propositional variables.

1. Show $(A \leftrightarrow (B \wedge C)) \equiv (A \vee \neg B \vee \neg C) \wedge (\neg A \vee B) \wedge (\neg A \vee C)$.
2. Show $(A \leftrightarrow \neg B) \equiv (A \vee B) \wedge (\neg A \vee \neg B)$.
3. Show $(A \leftrightarrow (B \to C)) \equiv (\neg A \vee \neg B \vee C) \wedge (A \vee B) \wedge (A \vee \neg C)$.

Exercise 13: Normal forms

Let $F = \neg((\neg A \to B) \wedge ((A \wedge \neg C) \to B))$ be a propositional formula over the propositional variables A, B, C.

1. Give the truth table of F and use it to construct formulae φ_c and φ_d in the canonical conjunctive and disjunctive normal form, respectively, that are equivalent to F.
2. Obtain a formula η_c in conjunctive normal form that is equivalent to F, using the equational transformations.