Logic for Computer Science

12 – Review

Wouter Swierstra

University of Utrecht

Last time

Inductive definitions

This lecture

Review of material covered in the mid-term

Review

- Introduction
- Propositions
- Sets
- Boolean algebra
- Predicate logic
- Proof strategies
- Functions
- Relations

Exam practical matters

Come on time

I will provide you with scrap paper.

Be sure to bring a photo ID, Solis login credentials, and a pen.

The exam lasts for 2 hours, which should be plenty of time.

Extra time students will have an extra 20 minutes.

There is also a ‘Minder Massaal’ room – contact me directly for more information.

Introduction

- What is logic?
- Why study logic?
- What is an invariant?

Propositions – material

- Propositional logic formulas
- Truth tables
- Proofs and properties (commutativity, associativity, de Morgan's laws, etc.)

Propositions – questions

Possible questions may include:

- Complete this truth table. . .
- Understand the syntax: draw a syntax tree, distinguish the precedence of operators. . ..
- Prove these two formulas are equivalent. . ..

Propositions – questions

Possible questions may include:

- Complete this truth table. . .
- Understand the syntax: draw a syntax tree, distinguish the precedence of operators. . ..
- Prove these two formulas are equivalent. . ..

But a solid understanding of propositional logic is necessary to understand the more complicated proofs in later chapters.

Don't worry about memorizing *all* possible laws about propositional logic. It can be useful to recognise de Morgan's laws, contraposition, etc. – but you won't need to reproduce them on the exam.

Sets – material

- Notions such as sets, elements, singletons, empty set, cardinality, powersets, . . .
- Various operations for combining sets, such as unions, complement, intersections, cartesian products, . . .
- Venn diagrams
- The subset relation, $A \subseteq B$ when $\forall x \left(x \in A \Rightarrow x \in B\right)$
- Equality between sets, A = B if and only if $\forall x \left(x \in A \Leftrightarrow x \in B\right)$

Sets – questions

- Prove that the sets X and Y are equal.
- Draw or interpret a Venn diagram.
- Model some structure as a set – similar to the model of the computer screen that we saw in the lectures.

Boolean algebra and circuits – material

- Boolean algebras & their properties
- Duality
- Circuit diagrams & their relation with boolean algebras
- Binary numbers and adders

Boolean algebras – questions

- Understand or optimize a given circuit;
- Apply duality;
- Add or convert binary numbers.
- Prove an equality in any boolean algebra;

I understand that proofs of theorems for boolean algebras are not always ‘intuitive’ – and require certain creative steps.

On the exam, I typically provide enough hints to point you in the right direction: use distributivity and the fact that $x + x = x$ to prove Y.

Laws of boolean algebras

You will need to know the 10 (or five) laws that form the definition of a boolean algebra.

If you need any auxiliary theorem/result to complete a proof, I will provide you with this in the question.

So there is no need to memorise every result in this chapter.

Predicate logic – material

- Predicates, universal and existential quantifier.

Predicate logic – material

- Predicates, universal and existential quantifier.

But also. . .

- Scope, free variables and bound variables
- Modelling natural language statements using predicate logic
- Laws for manipulating formulas in predicate logic

Predicate logic – questions

- Understand a predicate logic formula;
- Formalize some notion from some domain – such as a family tree – using predicate logic;
- Use the rules for manipulating formulas using predicate logic to prove two formulas equivalent:

$$\neg\forall x\ P(x) \Leftrightarrow \exists x\ \neg P(x)$$

$$\forall x\ (P(x) \wedge Q(x)) \Leftrightarrow (\forall x\ \ P(x)) \wedge (\forall x\ Q(x))$$

$$\ldots$$

Modelling with predicate logic

Given some description in natural language, such as the specification of a Sudoku puzzle, how to I turn this into a formula in predicate logic?

Modelling with predicate logic

Given some description in natural language, such as the specification of a Sudoku puzzle, how to I turn this into a formula in predicate logic?

There is unfortunately no ‘recipe’ of steps that I can give you that always works.

- Study examples, such as those covered in class or in the book;
- Do you want to make a statement about all things? Then typically start with a universal quantifier;
- Do you want to make a statement about some thing? Then typically start with a existential quantifier;
- If the statement makes some conditions, $\forall n > 3$ the property $P(n)$ holds – this is typically translated to logical implication.
- Try to break the statement into smaller pieces;
- Re-use other predicates/formulas you may have already defined (such as using the Sister(x,y) predicate to define Aunt(x,y)).

Proof strategies

- Introduction and elimination strategies for logical operators
- Introduction and elimination strategies for quantifiers
- ‘Derived’ proof strategies, such as contraposition, that can be justified using these strategies.

Proof strategies – questions

- What is the introduction/elimination strategy for X?
- Identify the proof strategies used in this proof;
- Which step in the following proof is wrong?
- **Writing proofs using strategies** - practice with the extra exercises!

How to write proofs

Once again, it is undecidable in general how to prove a given formula in predicate logic – there's no recipe I can give you.

For most of the exercises, however, the following approach can help:

- Write down precisely as possible what your assumptions are and what you are trying to prove.
- Try to apply the introduction strategies on the goal you are trying to prove.
- If you're lucky, this is all you need to do; sometimes – for example when you need to prove a conjunction – this breaks the problem into smaller pieces.
- If you get stuck, look at your assumptions. What elimination strategy can you apply to your assumptions? Does this teach you anything new?
- There's no shame in being stuck – some proofs require creativity!

Disjunction elimination

One of the hardest strategies to understand is that of disjunction elimination.

Question

Prove that for all sets A and B, $A \cup B = B \cup A$.

Or slightly harder:

Prove that for all sets A and B, $A \cup (B \cap C) \subseteq (A \cup B) \cap (A \cup C)$

Existence proofs

- Existential introduction is fairly straightforward: to proven $\exists x\ P(x)$ you get to invent some element a, but need to show that $P(a)$.

  Example $\exists x\ x - 7 = 0$ (and variations on this theme using quadratic equations in the book).

- Existential elimination is harder: if you know $\exists x\ P(x)$ how can you use this?

  You're allowed to assume that there is some arbitary a for which $P(a)$ holds. . .

  . . . but you don't know anything about a except that $P(a)$ holds.

Careful with assumptions; careful with names

A lot of the proof strategies start with 'Assume x is arbitrary. . .'

It is left a bit implicit that this x has to be a fresh name, that is not used anywhere else in the proof!

Similarly, be careful about the *scope* of your assumptions. They should never escape the surrounding 'box'.

Functions

- Definition of a function;
- Graphs of a function;
- Function composition;
- Function properties, such as injections, surjections and bijections;
- Using functions to compare the size of (infinite) sets;
- (6.4 - you should know about countable and uncountable sets - but no need to study the (proof of) Schröder-Bernstein Theorem)
- (6.5 is not part of the exam - on the Knaster-Tarski Theorem)

How can I show that the function $f : A \longrightarrow B$ is surjective?

How can I show that the function $f : A \rightarrow B$ is surjective?

We need to show:

$$\forall b \in B \quad \exists a \in A \quad f(a) = b$$

Prove that for every $b \in B$ there is some $a \in A$ such that $f(a) = b$. In other words – find an input mapped to every possible element of B.

Example: the length mapping Strings to Integers is surjective because for every n, the string consisting of repeating the character 'a' n times has length n.

Properties of functions and relations

How can I show that the function $f : A \longrightarrow B$ is injective?

Properties of functions and relations

How can I show that the function $f : A \to B$ is injective?

We need to show:

$$\forall a \in A \quad \forall a' \in A \quad f(a) = f(a') \Rightarrow a = a'$$

Assume that for some arbitrary a and a', we have that $f(a) = f(a')$. From this assumption, show that $a = a'$.

Example: The doubling function on natural numbers is injective – there are no two numbers that get mapped to the same double.

But the $\sin : \mathbb{R} \to [0, 1]$ is not injective – it crosses 0 infinitely many times.

Relations

- Definition of a relation;
- Composing, inverting, unions or intersections of relations
- Properties of relations: reflexive, irreflexive, symmetric, antisymmetric, asymmetric, transitive, . . .
- Equivalence relations, equivalence classes and partitions
- (Partial orders and total orders – covered in the book, not in the slides and will not be on the exam)

Properties of functions and relations

How can I show that the relation R is transitive?

How can I show that the relation R is transitive?

Follow the definition and the corresponding proof strategies:

Assume that xRy and yRz. Can we prove xRz?

Example The subset relation, $\subseteq$, is transitive.

Material

Modelling Computing Systems Chapter 1-7 with the exception of 6.5

Good luck on your mid-term exam!