Coursework

- Write essay.
- Choice of five topics.
- Submit by 16:15 on May 13 in Departmental Office (1 West 2.23).
- Proportion of unit assessment: 25%.
- **Individual coursework**, i.e. complete it on your own.

Five topics

- **Undecidability of predicate logic.** Explained later in this lecture.
- **Verification by model checking.** This is about scenarios where we want to check automatically if $M \models A$ for some reactive or concurrent system M and some modal or temporal formula A describing a desirable property.

Five topics

- **Compactness, and Löwenheim-Skolem theorems.** This is about the connection between sets Γ of formulæ and their models.
- **Gödel's incompleteness theorems.** Famous results about the limits of deduction in predicate logic.
- **Fuzzy logic and its applications.** Reasoning about "partial truth". (Fairly easy subject; compensate by deep literature search.)

Reference on writing essays

Michael Alley, The Craft of Scientific Writing, Prentice-Hall, 1987.

Approximate structure

(15%) A brief introduction to the topic: its scope and significance.

(30%) An explanation of the main ideas, including their conceptual basis and technical development.

(30%) An explanation of the applications of the main ideas. For example, logic itself, or computing, AI, or mathematics.

(20%) A conclusion summarizing the current state of the topic. What are current issues? Are new applications emerging? (These two question may not apply to essays of a more theoretical nature, resulting in a greater weighting of other aspects.) What do you think about the things you have described?

(5%) A list of references; specific references and citations should be given in the style used in the course book (Huth and Ryan).

Summary of quantifier rules

The introduction and elimination rules for quantifiers are

$$\frac{\Gamma \vdash A}{\Gamma \vdash \forall x.A}\forall i \quad \text{if } x \notin FV(\Gamma) \qquad\qquad \frac{\Gamma \vdash \forall x.A}{\Gamma \vdash A[t/x]}\forall e$$

$$\frac{\Gamma \vdash A[t/x]}{\Gamma \vdash A}\exists i \qquad\qquad \frac{\Gamma \vdash \exists x.A \quad \Gamma, A \vdash B}{\Gamma \vdash B}\exists e \quad \text{if } x \notin FV(\Gamma \cup \{B\}),$$

where for $\forall e$ and $\exists i$, the term t must be free for x in A.

Soundness

(This slide and the next three are an improved version of the slides about soundness in the previous handout.)
Theorem.[Soundness] If $\Gamma \vdash A$, then $\Gamma \models A$.

- The soundness of the rules for $\wedge$, $\rightarrow$, $\perp$, and $\vee$ is shown in the same way as for propositional logic.
- So it remains to show the soundness of $\forall i$, $\forall e$, $\exists i$, and $\exists e$.

Soundness of $\forall i$ and $\exists e$

The soundness proof for $\forall i$ works as follows: suppose that $\Gamma \models A$ and $M \models \Gamma$. To see that $M \models \forall x.A$, we need to show that $M[a/x] \models A$ for all $a \in U$. Because $M \models \Gamma$ and x does not occur freely in Γ, we have $M[a/x] \models \Gamma$. Because $\Gamma \models A$, we get $M[a/x] \models A$.

Exercise: Prove the soundness of $\exists e$.

Soundness of $\forall e$ and $\exists i$

The soundness proof for these two rules requires the following lemma, which can be proved by induction on A.

Lemma. For every formula A, every term t which is free for x in A, and every situation M, it holds that

$$M \models A[t/x] \quad \text{iff} \quad M[[\![t]\!]_M/x] \models A.$$

Soundness of $\forall e$ and $\exists i$

The soundness proof for $\forall e$ works as follows: suppose that $\Gamma \models \forall x.A$, and let t be free for x in A. To see that $\Gamma \models A$, suppose that $M \models \Gamma$. Because $\Gamma \models \forall x.A$, we have $M[a/x] \models A$ for all $a \in U$. In particular, $M[[\![t]\!]_M/x] \models A$. By the lemma, this is so iff $M \models A[t/x]$.

Exercise: Prove the soundness of $\exists i$.

Completeness

Theorem.[Completeness] If $\Gamma \models A$, then $\Gamma \vdash A$ is provable in ND.

- The completeness proof follows the same scheme as the one for propositional logic.
- Only the Model Existence Lemma needs to be re-proved, because situations now involve a universe, functions, and predicates.
- The proof of the MEL is still based on (an updated version of) maximally consistent sets. (For details, see van Dalen.)

Undecidability of predicate logic

Recall that the algorithm below can be used to decide the validity of $A_1, \ldots, A_n \models B$ in propositional logic.

1. Check for every situation M if, whenever $M \models A_i$ for all $i \in \{1, \ldots, n\}$, then $M \models B$.
2. If this is true, then $A_1, \ldots, A_n \models B$,
3. otherwise $A_1, \ldots, A_n \not\models B$.

Undecidability of predicate logic

- In contrast to propositional logic, predicate logic has infinitely many situations (because universes can have arbitrary size).
- So the instructions above no longer work—there are always more models to check.

Undecidability of predicate logic

- Fortunately, by soundness and completeness, we can check $A_1, \ldots, A_n \vdash B$ instead of $A_1, \ldots, A_n \models B$.
- ND proofs (which can be represented by strings) are **recursively enumerable**, i.e. there is an enumeration

$$\Phi_1, \Phi_2, \Phi_3, \ldots$$

of all proofs, and it can be implemented by a program that sends a positive integer to a string representing a proof.

Undecidability of predicate logic

So the following algorithm succeeds if $A_1, \ldots, A_n \models B$:

i = 0;

success = false;

while(success = false) {

if the proof Φ_i shows $A_1, \ldots, A_n \vdash B$ then

success = true;

i = i+1;

}

Undecidability of predicate logic

- The algorithm gives a positive answer if $A_1, \ldots, A_n \models B$.
- But goes into in infinite loop if $A_1, \ldots, A_n \not\models B$.
- Can this be fixed, i.e. is there an algorithm that also comes back with an answer if $A_1, \ldots, A_n \not\models B$?
- The following theorem states that this is impossible.

Undecidability of predicate logic

Theorem. There is no algorithm that, given any formula A of predicate logic, decides whether A is valid or not.

There are various proofs of the undecidability theorem (see e.g. Huth/Ryan, Boolos/Burgess/Jeffrey). This is one of the possible coursework essays.

First-order logic

Predicate logic is also called **first-order logic**. This terminology refers to the types of the variables. For example,

$$\exists f.\forall x.f(x) = x$$

is a **second-order formula**, because f ranges over functions $U \to U$, not elements of U. A formula is third-order if it contains quantifies ranging over things of type $(U \to U) \to U$, and so on.