Proof editor for natural deduction

Keywords

proof editor, propositional logic, predicate logic, natural deduction

Background

We are all familiar with formulas in propositional or predicate logic like

$p \wedge q \rightarrow r \vee \neg q \rightarrow r$

or

$\forall x. \neg P(x) \rightarrow \neg \exists x. P(x)$

In discrete mathematics we study the validity of the formulas by understanding the behavior of the connectives and quantifiers, and giving an interpretation to the symbols of the language.

Then a formula $p \wedge q$ will be true if both p and q are true.
A formula $\forall x. P(x)$ will be true in a particular interpretation whenever any element in the domain we work with satisfies the interpretation of the predicate P in that domain.

This is referred to as the semantics of propositional and predicate logic.

But both propositional and predicate logic can be used as a formal language.
Here, we have inference rules that tell us how to prove formulas that contain a certain connective/quantifier and how to use formulas that contain a certain connective/quantifier to prove other formulas.

Then we can have a proof of the formula $p \wedge q$ whenever we have a proof of both p and q.
To have a proof of the formula $\forall x. P(x)$ we need to have a proof of $P(x0)$ for any possible element x0.

Project description

This project is about constructing a proof editor that allows to prove formulas in both propositional and predicate logic following a natural deduction style as presented in form example in chapters 1 and 2.1-2.4 of Logic in Computer Science by Michael Huth and Mark Ryan, second edition (there used to be an electronic version available at Chalmers' library).

The proof editor should not only handle error messages, the interface with the user, the graphical representation of the proofs, but also give hints on the rules that could be applicable when constructing a specific proof.

Suggested reading
https://en.wikipedia.org/wiki/First-order_logic
https://en.wikipedia.org/wiki/Natural_deduction
http://www.cse.chalmers.se/~hallgren/Alfa/

https://proofs.openlogicproject.org/
https://gupea.ub.gu.se/handle/2077/69606
https://odr.chalmers.se/handle/20.500.12380/251213

Target group

DV, IT, D, TM, F, Z (actually anyone with knowledge in logic)

The final report will be written in Swedish

Special Pre-requisites

Students should be familiar basic concepts in logic at the level taught in a discrete mathematics course. Good programming skills are necessary.

It is an advantage, but not necessary, if students are familiar more advance logic (for example via the course Logic in Computer science).

Proposal author

Ana Bove (bove@chalmsers.se)

Supervisor

Possibly Ana Bove but also many others at the LT unit or CS division.