Properties, Model, Correctness

- Program correctness.
- Syntax.
- Semantics.
- Substitutive model of computation.

How do we formally verify the program correctness?

Propositions and Proofs

What is a proposition

- Sky is blue;
- Snow is white;
- Pigs are cute;

Collins Dictionary has a definition for proposition as "a statement or an idea that people can consider or discuss whether it is true."

Propositions

- Statements about the program model.
- In general, provable and refutable.
- Focus on Equivalences.

Formally, propositions are often modeled as functions which map a possible world to a truth value.

Propositions and Proofs

Proofs

- Evaluation following small steps operational semantics.
- How did you perform the proofs?

Claim: $(+\ 1\ 2\ 3) \equiv (/\ (*\ 3\ (+\ 3\ 1))\ 2)$.

Symbolic Execution

- $2 \equiv 2$
- $a \equiv b$

Sets of cases VS Concrete cases

- Part of the evaluation are unknown.
- Contains variables.
- Group of potential behaviors/instances.

Abstract Propositions

- Existential quantification.
- Universal quantification.
- Adding quantification makes proposition provable.

Symbolic Execution

Claim: for all numbers n, (square n) $\equiv$ (* n n).

- Consider variables constant/value, unknown.
- We prove over group of objects (sets).

Pattern match for case analysis, intricate proofs.

Claim: for all booleans b, (my-and b #f) $\equiv$ #f.

```
match b with
  | #t -> (if #t #f #f) == #f
  | #f -> (if #f #f #f) == #f
```

Q & A