Program Verification

- Program model includes syntax and semantics.
- Propositions, and how to perform proof.

How do we verify the program that is correct?

Preconditions and Postconditions

$$(> (+\ b\ 1)(\times\ a\ 4))$$

- Informally, preconditions as assumptions, postcondtions as conclusions after execution of the program.
- Formally, both preconditions and postconditions are constraints telling what programs are supposed to behave before and after the execution.
- $(<$ x y$) \equiv$ true $\rightarrow$ $(<$ x y$)$

$$\{Preconditions\}\ Program\ \{Postconditions\}$$

Beauty of Mathematics

- General;
- Precise;
- Clean;

```
;;; Returns the (0-based) index of element x in list l.
 (index-of x l)


  ;;; Returns the length of list l.
 (list-length l)
```

Tracking and Ulilizing Assumptions

- Programmers focus on implementing core functionality.
- Verify program correctness requires more efforts, writing down preconditions and positions explicitly, precisely including all possible cases.

$$\{Preconditions\}\ Program\ \{Postconditions\}$$

Accumulate constraints when evaluate program expressions step by step, constraints are abstract propositions with variable id, we call program states at current location.

Q & A