Program Verification

- Program correctness ≡ Functional Correctness assuming termination
  + Termination
- Proof for functional correctness assuming termination called Partial Correctness

  Partial Correctness + Proof for termination ≡ Total Correctness

- Program ≡ Seq. of statements of type

(i) If B then S_1 else S_2 (if-then-else statement)

B ≡ predicate over prog. variables, known as gaurd

S_i ≡ simple assignment of type, $\vec{x} \Leftarrow f_i(\vec{x})$

(vector of prog. variables $\vec{x}$ assigned the value $f_i(\vec{x})$)

special cases: S_2 is vacuous (If B then S_1) ⇒ $f_2(\vec{x}) = \vec{x}$.

B is vacuous (S_1) ⇒ B ≡ TRUE

(ii) While B do S (while-loop)

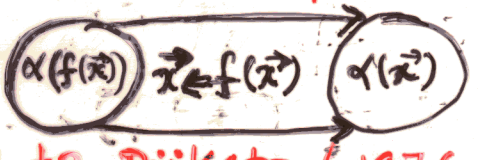
S executes as long as predicate/gaurd B is satisfied.

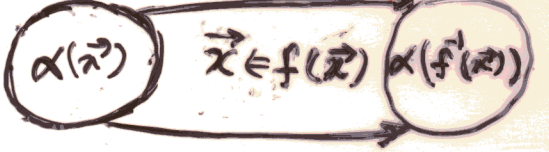
Inference-rules for simplest assignment

(i) $\dfrac{}{\alpha(f(\vec{x}))\{\vec{x} \Leftarrow f(\vec{x})\}\alpha(\vec{x})}$ OR $\dfrac{}{\alpha(\vec{x})\{\vec{x} \Leftarrow f(\vec{x})\}\alpha(f^{-1}(\vec{x}))}$

$\alpha(f(\vec{x}))$: weakest precondition

$\alpha(f^{-1}(\vec{x}))$: Strongest postcond.





Due to Dijkstra (1976 book)

Example: $odd(x+1)\{x \Leftarrow x+1\}\ odd(x) \equiv even(x)\{x \Leftarrow x+1\}\ odd(x)$

$odd(x)\{x \Leftarrow x+1\}\ odd(x+1) \equiv odd(x)\{x \Leftarrow x+1\}\ even(x)$

Note: $f(x) = x+1 \Rightarrow f^{-1}(x) = x-1$