# Program verification: Framework

CS242 Formal Specification and Verification

University of Warwick

Autumn term 2006

# Programming language

$$
\begin{array}{lcl}
E & ::= & n \mid x \mid (-E) \mid (E + E) \mid (E - E) \mid (E * E) \\
B & ::= & \texttt{true} \mid \texttt{false} \mid (!B) \mid (B \& B) \mid (B \parallel B) \mid (E < E) \\
C & ::= & x = E \mid C; C \mid \texttt{if}\, B\, \{C\}\, \texttt{else}\, \{C\} \mid \texttt{while}\, B\, \{C\}
\end{array}
$$

Negation $(-E)$ binds more tightly than $*$, which binds more tightly than $+$ and $-$.

# Hoare triples

$$(\!|\phi|\!)\ P\ (\!|\psi|\!)$$

- $P$ is a command.
- $\phi$ is called the precondition, and $\psi$ the postcondition.
- No variable which occurs in $P$ should occur bound in $\phi$ or $\psi$.
- Variables which occur free in $\phi$ or $\psi$, but do not occur in $P$ are called logical variables.

A state $I$ is a function which assigns to each variable $x$ an integer $I(x)$.
We write $I \models \phi$ iff $\mathcal{M} \models_I \phi$, where $\mathcal{M}$ is the standard model whose universe is the set of all integers.

## Partial correctness

$\models_{\mathsf{par}} (\!|\phi|\!)\ P\ (\!|\psi|\!)$ iff, for all $I$ which satisfy $\phi$, if $P$ terminates from $I$ and results in $I'$, then $I'$ satisfies $\psi$.

## Total correctness

$\models_{\mathsf{tot}} (\!|\phi|\!)\ P\ (\!|\psi|\!)$ iff, for all $I$ which satisfy $\phi$, $P$ does terminate from $I$ and results in some $I'$ which satisfies $\psi$.