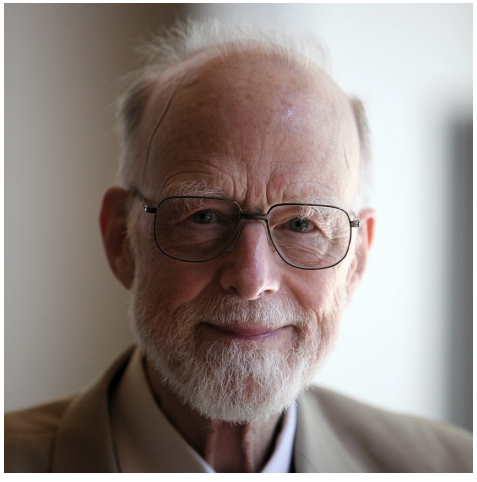
COMP2111 Week 8
Term 1, 2024
Hoare Logic

Sir Tony Hoare

- Pioneer in formal verification
- Invented: Quicksort,
- the null reference (called it his “billion dollar mistake”)
- CSP (formal specification language), and
- Hoare Logic



Summary

- $\mathcal{L}$: A simple imperative programming language
- Hoare triples (SYNTAX)
- Hoare logic (PROOF)
- Semantics for Hoare logic

Summary

- $\mathcal{L}$: A simple imperative programming language
- Hoare triples (SYNTAX)
- Hoare logic (PROOF)
- Semantics for Hoare logic

Imperative Programming

imperō

Definition

Imperative programming is where programs are described as a series of *statements* or commands to manipulate mutable *state* or cause externally observable *effects*.

States may take the form of a *mapping* from variable names to their values, or even a model of a CPU state with a memory model (for example, in an *assembly language*).

$\mathcal{L}$: A simple imperative programming language

Consider the vocabulary of basic arithmetic:

- Constant symbols: $0, 1, 2, \ldots$
- Function symbols: $+, *, \ldots$
- Predicate symbols: $<, \leq, \geq, |, \ldots$

$\mathcal{L}$: A simple imperative programming language

Consider the vocabulary of basic arithmetic:

- Constant symbols: $0, 1, 2, \ldots$
- Function symbols: $+, *, \ldots$
- Predicate symbols: $<, \leq, \geq, |, \ldots$
- An **(arithmetic) expression** is a term over this vocabulary.

$\mathcal{L}$: A simple imperative programming language

Consider the vocabulary of basic arithmetic:

- Constant symbols: $0, 1, 2, \ldots$
- Function symbols: $+, *, \ldots$
- Predicate symbols: $<, \leq, \geq, |, \ldots$
- An **(arithmetic) expression** is a term over this vocabulary.
- A **boolean expression** is a predicate formula over this vocabulary.

The language $\mathcal{L}$

The language $\mathcal{L}$ is a simple imperative programming language made up of four statements:

Assignment: $x := e$

where x is a variable and e is an arithmetic expression.

The language $\mathcal{L}$

The language $\mathcal{L}$ is a simple imperative programming language made up of four statements:

Assignment: $x{:=}e$

where x is a variable and e is an arithmetic expression.

Sequencing: $P;Q$

The language $\mathcal{L}$

The language $\mathcal{L}$ is a simple imperative programming language made up of four statements:

Assignment: $x{:=}e$

where x is a variable and e is an arithmetic expression.

Sequencing: $P;Q$

Conditional: if g then P else Q fi

where g is a boolean expression.

The language $\mathcal{L}$

The language $\mathcal{L}$ is a simple imperative programming language made up of four statements:

Assignment: $x{:=}e$

where x is a variable and e is an arithmetic expression.

Sequencing: $P;Q$

Conditional: if g then P else Q fi

where g is a boolean expression.

While: while g do P od

Factorial in $\mathcal{L}$

Example

$i := 0;$
$m := 1;$
while $i < N$ do
  $i := i + 1;$
  $m := m * i$
od

Summary

- $\mathcal{L}$: A simple imperative programming language
- Hoare triples (SYNTAX)
- Hoare logic (PROOF)
- Semantics for Hoare logic

Summary

- $\mathcal{L}$: A simple imperative programming language
- Hoare triples (SYNTAX)
- Hoare logic (PROOF)
- Semantics for Hoare logic

Hoare Logic

We'll define a *Hoare Logic* for $\mathcal{L}$ to allow us to prove properties of our program.
We write a *Hoare triple* judgement as:

$$\{\varphi\}\ P\ \{\psi\}$$

Where φ and ψ are logical formulae about states, called *assertions*, and $P \in \mathcal{L}$. This triple states that if the program P terminates successfully from a starting state satisfying the *precondition* φ, then the final state will satisfy the *postcondition* ψ.

Hoare triple: Examples

Example

$$\{(x = 0)\}\, x := 1\, \{(x = 1)\}$$

Hoare triple: Examples

Example

$$\{(x = 0)\}\, x := 1\, \{(x = 1)\}$$

$$\{(x = 499)\}\, x := x + 1\, \{(x = 500)\}$$

Hoare triple: Examples

Example

$$\{(x = 0)\}\, x := 1\, \{(x = 1)\}$$

$$\{(x = 499)\}\, x := x + 1\, \{(x = 500)\}$$

$$\{(x > 0)\}\, y := 0 - x\, \{(y < 0) \wedge (x \neq y)\}$$

Hoare triple: Factorial Examples

Example

$$
\begin{array}{l}
\{N \geq 0\} \\
i := 0; \\
m := 1; \\
\text{while } i < N \text{ do} \\
\quad i := i + 1; \\
\quad m := m * i \\
\text{od} \\
\{m = N!\}
\end{array}
$$

Summary

- $\mathcal{L}$: A simple imperative programming language
- Hoare triples (SYNTAX)
- Hoare logic (PROOF)
- Semantics for Hoare logic

Motivation

Question

We know what we want informally; how do we establish when a triple is valid?

Motivation

Question

We know what we want informally; how do we establish when a triple is valid?

- Develop a semantics, OR

Hoare logic consists of one axiom and four inference rules for deriving Hoare triples.

Motivation

Question

We know what we want informally; how do we establish when a triple is valid?

- Develop a semantics, OR
- Derive the triple in a syntactic manner (i.e. Hoare proof)

Hoare logic consists of one axiom and four inference rules for deriving Hoare triples.

Assignment

$$\frac{}{\{\varphi[e/x]\}\, x := e\, \{\varphi\}} \quad \text{(assign)}$$

Intuition:
If x has property φ *after* executing the assignment; then e must have property φ *before* executing the assignment

Assignment: Example

Example

$$\{(y = 0)\}\, x := y\, \{(x = 0)\}$$

Assignment: Example

Example

$$\{(y = 0)\}\, x := y\, \{(x = 0)\}$$

$$\{\qquad\qquad\}\, x := y\, \{(x = y)\}$$

Assignment: Example

Example

$$\{(y = 0)\}\, x := y\, \{(x = 0)\}$$

$$\{(y = y)\}\, x := y\, \{(x = y)\}$$

Assignment: Example

Example

$$\{(y = 0)\}\, x := y\, \{(x = 0)\}$$

$$\{(y = y)\}\, x := y\, \{(x = y)\}$$

$$\{\qquad\qquad\}\, x := 1\, \{(x < 2)\}$$

Assignment: Example

Example

$$\{(y = 0)\}\, x := y\, \{(x = 0)\}$$

$$\{(y = y)\}\, x := y\, \{(x = y)\}$$

$$\{(1 < 2)\}\, x := 1\, \{(x < 2)\}$$

$$\{(y = 3)\}\, x := y\, \{(x > 2)\}$$

Assignment: Example

Example

$$\{(y = 0)\}\, x := y\, \{(x = 0)\}$$

$$\{(y = y)\}\, x := y\, \{(x = y)\}$$

$$\{(1 < 2)\}\, x := 1\, \{(x < 2)\}$$

$$\{(y = 3)\}\, x := y\, \{(x > 2)\} \qquad \textit{Problem}!$$

Sequence

$$\frac{\{\varphi\}\, P\, \{\psi\} \qquad \{\psi\}\, Q\, \{\rho\}}{\{\varphi\}\, P; Q\, \{\rho\}} \quad \text{(seq)}$$

Intuition:
If the postcondition of P matches the precondition of Q we can sequentially combine the two program fragments

Sequence: Example

Example

$$\frac{\{\qquad\}\, x := 0\, \{\qquad\} \qquad \{\qquad\}\, y := 0\, \{(x = y)\}}{\{\qquad\}\, x := 0; y := 0\, \{(x = y)\}}\ (\text{seq})$$

Sequence: Example

Example

$$\frac{\{\quad\}\, x := 0\, \{(x = 0)\} \qquad \{(x = 0)\}\, y := 0\, \{(x = y)\}}{\{\quad\}\, x := 0; y := 0\, \{(x = y)\}}\ (\text{seq})$$

Sequence: Example

Example

$$\frac{\{(0=0)\}\, x := 0\, \{(x=0)\} \qquad \{(x=0)\}\, y := 0\, \{(x=y)\}}{\{(0=0)\}\, x := 0; y := 0\, \{(x=y)\}}\ (\text{seq})$$

Conditional

$$\frac{\{\varphi \wedge g\}\, P\, \{\psi\} \qquad \{\varphi \wedge \neg g\}\, Q\, \{\psi\}}{\{\varphi\}\ \text{if } g \text{ then } P \text{ else } Q \text{ fi}\ \{\psi\}} \quad \text{(if)}$$

Intuition:

- When a conditional is executed, either P or Q will be executed.
- For the postcondition ψ to be established, *either* branch must terminate in a state satisfying ψ.

While

$$\frac{\{\varphi \wedge g\}\, P\, \{\varphi\}}{\{\varphi\}\, \text{while } g \text{ do } P \text{ od}\, \{\varphi \wedge \neg g\}} \quad \text{(loop)}$$

Intuition:

- φ is a **loop invariant**. It must be both a pre- and postcondition of P, so that sequences of Ps can be run together.
- If the while loop terminates, g cannot hold.

Consequence

There is one more rule, called the *rule of consequence*, that we need to insert ordinary logical reasoning into our Hoare logic proofs:

$$\frac{\varphi' \to \varphi \qquad \{\varphi\}\, P\, \{\psi\} \qquad \psi \to \psi'}{\{\varphi'\}\, P\, \{\psi'\}} \quad \text{(cons)}$$

Consequence

There is one more rule, called the *rule of consequence*, that we need to insert ordinary logical reasoning into our Hoare logic proofs:

$$\frac{\varphi' \to \varphi \qquad \{\varphi\}\, P\, \{\psi\} \qquad \psi \to \psi'}{\{\varphi'\}\, P\, \{\psi'\}} \quad \text{(cons)}$$

Intuition:

- Adding assertions to the precondition makes it more likely the postcondition will be reached
- Removing assertions from the postcondition makes it more likely the postcondition will be reached
- If you can reach the postcondition initially, then you can reach it in the more likely scenario

Back to Assignment Example

Example

$$\{(y = 3)\}\, x := y\, \{(x > 2)\} \qquad \textit{Problem}!$$

Back to Assignment Example

Example

$$\{(y = 3)\}\, x := y\, \{(x > 2)\} \qquad \textit{Problem}!$$

$$\{(y > 2)\} x := y \{(x > 2)\} (\textit{assign})$$

Back to Assignment Example

Example

$$\{(y = 3)\}\, x := y\, \{(x > 2)\} \qquad \textit{Problem}!$$

$$\{(y = 3)\}x := y\{(x > 2)\}(\textit{assign}, \textit{cons})$$
$$\{(y > 2)\}x := y\{(x > 2)\}(\textit{assign})$$

Factorial Example

Let's verify the Factorial program using our Hoare rules:

$\{N \geq 0\}$

$i := 0;$
$m := 1;$

while $i < N$ do

$i := i + 1;$

$m := m \times i$

od
$\{m = N!\}$

$$\frac{\{\varphi \wedge g\}\ P\ \{\psi\} \quad \{\varphi \wedge \neg g\}\ Q\ \{\psi\}}{\{\varphi\}\ \texttt{if}\ g\ \texttt{then}\ P\ \texttt{else}\ Q\ \texttt{fi}\ \{\psi\}}$$

$$\frac{}{\{\varphi[x := e]\}\ x := e\ \{\varphi\}}$$

$$\frac{\{\varphi \wedge g\}\ P\ \{\varphi\}}{\{\varphi\}\ \texttt{while}\ g\ \texttt{do}\ P\ \texttt{od}\ \{\varphi \wedge \neg g\}}$$

$$\frac{\{\varphi\}\ P\ \{\alpha\} \qquad \{\alpha\}\ Q\ \{\psi\}}{\{\varphi\}\ P; Q\ \{\psi\}}$$

$$\frac{\varphi' \Rightarrow \varphi \qquad \{\varphi\}\ P\ \{\psi\} \qquad \psi \Rightarrow \psi'}{\{\varphi'\}\ P\ \{\psi'\}}$$

Factorial Example

Let's verify the Factorial program using our Hoare rules:

$\{N \geq 0\}$

$i := 0;$
$m := 1;$

while $i < N$ do

$i := i + 1;$

$m := m \times i$

od $\{m = i! \wedge N \geq 0 \wedge i = N\}$
$\{m = N!\}$

$$\frac{\{\varphi \wedge g\}\ P\ \{\psi\} \quad \{\varphi \wedge \neg g\}\ Q\ \{\psi\}}{\{\varphi\}\ \texttt{if}\ g\ \texttt{then}\ P\ \texttt{else}\ Q\ \texttt{fi}\ \{\psi\}}$$

$$\frac{}{\{\varphi[x := e]\}\ x := e\ \{\varphi\}}$$

$$\frac{\{\varphi \wedge g\}\ P\ \{\varphi\}}{\{\varphi\}\ \texttt{while}\ g\ \texttt{do}\ P\ \texttt{od}\ \{\varphi \wedge \neg g\}}$$

$$\frac{\{\varphi\}\ P\ \{\alpha\} \qquad \{\alpha\}\ Q\ \{\psi\}}{\{\varphi\}\ P; Q\ \{\psi\}}$$

$$\frac{\varphi' \Rightarrow \varphi \qquad \{\varphi\}\ P\ \{\psi\} \qquad \psi \Rightarrow \psi'}{\{\varphi'\}\ P\ \{\psi'\}}$$

Factorial Example

Let's verify the Factorial program using our Hoare rules:

$\{N \geq 0\}$

$i := 0;$

$m := 1;$

$\{m = i! \wedge N \geq 0\}$

while $i < N$ do

$i := i + 1;$

$m := m \times i$

od $\{m = i! \wedge N \geq 0 \wedge i = N\}$

$\{m = N!\}$

$$\frac{\{\varphi \wedge g\}\ P\ \{\psi\} \quad \{\varphi \wedge \neg g\}\ Q\ \{\psi\}}{\{\varphi\}\ \texttt{if}\ g\ \texttt{then}\ P\ \texttt{else}\ Q\ \texttt{fi}\ \{\psi\}}$$

$$\frac{}{\{\varphi[x := e]\}\ x := e\ \{\varphi\}}$$

$$\frac{\{\varphi \wedge g\}\ P\ \{\varphi\}}{\{\varphi\}\ \texttt{while}\ g\ \texttt{do}\ P\ \texttt{od}\ \{\varphi \wedge \neg g\}}$$

$$\frac{\{\varphi\}\ P\ \{\alpha\} \qquad \{\alpha\}\ Q\ \{\psi\}}{\{\varphi\}\ P; Q\ \{\psi\}}$$

$$\frac{\varphi' \Rightarrow \varphi \qquad \{\varphi\}\ P\ \{\psi\} \qquad \psi \Rightarrow \psi'}{\{\varphi'\}\ P\ \{\psi'\}}$$

Factorial Example

Let's verify the Factorial program using our Hoare rules:

$$\{N \geq 0\}$$
$$i := 0;$$
$$m := 1;$$
$$\{m = i! \wedge N \geq 0\}$$
$$\text{while } i < N \text{ do}$$
$$i := i + 1;$$
$$m := m \times i$$
$$\{m = i! \wedge N \geq 0\}$$
$$\text{od } \{m = i! \wedge N \geq 0 \wedge i = N\}$$
$$\{m = N!\}$$

$$\frac{\{\varphi \wedge g\}\ P\ \{\psi\} \quad \{\varphi \wedge \neg g\}\ Q\ \{\psi\}}{\{\varphi\}\ \texttt{if}\ g\ \texttt{then}\ P\ \texttt{else}\ Q\ \texttt{fi}\ \{\psi\}}$$

$$\frac{}{\{\varphi[x := e]\}\ x := e\ \{\varphi\}}$$

$$\frac{\{\varphi \wedge g\}\ P\ \{\varphi\}}{\{\varphi\}\ \texttt{while}\ g\ \texttt{do}\ P\ \texttt{od}\ \{\varphi \wedge \neg g\}}$$

$$\frac{\{\varphi\}\ P\ \{\alpha\} \qquad \{\alpha\}\ Q\ \{\psi\}}{\{\varphi\}\ P; Q\ \{\psi\}}$$

$$\frac{\varphi' \Rightarrow \varphi \qquad \{\varphi\}\ P\ \{\psi\} \qquad \psi \Rightarrow \psi'}{\{\varphi'\}\ P\ \{\psi'\}}$$

Factorial Example

Let's verify the Factorial program using our Hoare rules:

$$\{N \geq 0\}$$
$$i := 0;$$
$$m := 1;$$
$$\{m = i! \wedge N \geq 0\}$$
$$\text{while } i < N \text{ do } \{m = i! \wedge N \geq 0 \wedge i < N\}$$
$$i := i + 1;$$
$$m := m \times i$$
$$\{m = i! \wedge N \geq 0\}$$
$$\text{od } \{m = i! \wedge N \geq 0 \wedge i = N\}$$
$$\{m = N!\}$$

$$\frac{\{\varphi \wedge g\}\ P\ \{\psi\} \quad \{\varphi \wedge \neg g\}\ Q\ \{\psi\}}{\{\varphi\}\ \texttt{if}\ g\ \texttt{then}\ P\ \texttt{else}\ Q\ \texttt{fi}\ \{\psi\}}$$

$$\frac{}{\{\varphi[x := e]\}\ x := e\ \{\varphi\}}$$

$$\frac{\{\varphi \wedge g\}\ P\ \{\varphi\}}{\{\varphi\}\ \texttt{while}\ g\ \texttt{do}\ P\ \texttt{od}\ \{\varphi \wedge \neg g\}}$$

$$\frac{\{\varphi\}\ P\ \{\alpha\} \qquad \{\alpha\}\ Q\ \{\psi\}}{\{\varphi\}\ P; Q\ \{\psi\}}$$

$$\frac{\varphi' \Rightarrow \varphi \qquad \{\varphi\}\ P\ \{\psi\} \qquad \psi \Rightarrow \psi'}{\{\varphi'\}\ P\ \{\psi'\}}$$

Factorial Example

Let's verify the Factorial program using our Hoare rules:

$$\{N \geq 0\}$$
$$i := 0;$$
$$m := 1;$$
$$\{m = i! \wedge N \geq 0\}$$
$$\text{while } i < N \text{ do } \{m = i! \wedge N \geq 0 \wedge i < N\}$$
$$\quad i := i + 1;$$
$$\quad \{m \times i = i! \wedge N \geq 0\}$$
$$\quad m := m \times i$$
$$\quad \{m = i! \wedge N \geq 0\}$$
$$\text{od } \{m = i! \wedge N \geq 0 \wedge i = N\}$$
$$\{m = N!\}$$

$$\frac{\{\varphi \wedge g\}\ P\ \{\psi\} \quad \{\varphi \wedge \neg g\}\ Q\ \{\psi\}}{\{\varphi\}\ \texttt{if}\ g\ \texttt{then}\ P\ \texttt{else}\ Q\ \texttt{fi}\ \{\psi\}}$$

$$\frac{}{\{\varphi[x := e]\}\ x := e\ \{\varphi\}}$$

$$\frac{\{\varphi \wedge g\}\ P\ \{\varphi\}}{\{\varphi\}\ \texttt{while}\ g\ \texttt{do}\ P\ \texttt{od}\ \{\varphi \wedge \neg g\}}$$

$$\frac{\{\varphi\}\ P\ \{\alpha\} \qquad \{\alpha\}\ Q\ \{\psi\}}{\{\varphi\}\ P; Q\ \{\psi\}}$$

$$\frac{\varphi' \Rightarrow \varphi \qquad \{\varphi\}\ P\ \{\psi\} \qquad \psi \Rightarrow \psi'}{\{\varphi'\}\ P\ \{\psi'\}}$$

Factorial Example

Let's verify the Factorial program using our Hoare rules:

$$\{N \geq 0\}$$
$$i := 0;$$
$$m := 1;$$
$$\{m = i! \wedge N \geq 0\}$$
$$\text{while } i < N \text{ do } \{m = i! \wedge N \geq 0 \wedge i < N\}$$
$$\quad \{m \times (i+1) = (i+1)! \wedge N \geq 0\}$$
$$\quad i := i + 1;$$
$$\quad \{m \times i = i! \wedge N \geq 0\}$$
$$\quad m := m \times i$$
$$\quad \{m = i! \wedge N \geq 0\}$$
$$\text{od } \{m = i! \wedge N \geq 0 \wedge i = N\}$$
$$\{m = N!\}$$

$$\frac{\{\varphi \wedge g\}\ P\ \{\psi\} \quad \{\varphi \wedge \neg g\}\ Q\ \{\psi\}}{\{\varphi\}\ \texttt{if}\ g\ \texttt{then}\ P\ \texttt{else}\ Q\ \texttt{fi}\ \{\psi\}}$$

$$\frac{}{\{\varphi[x := e]\}\ x := e\ \{\varphi\}}$$

$$\frac{\{\varphi \wedge g\}\ P\ \{\varphi\}}{\{\varphi\}\ \texttt{while}\ g\ \texttt{do}\ P\ \texttt{od}\ \{\varphi \wedge \neg g\}}$$

$$\frac{\{\varphi\}\ P\ \{\alpha\} \qquad \{\alpha\}\ Q\ \{\psi\}}{\{\varphi\}\ P; Q\ \{\psi\}}$$

$$\frac{\varphi' \Rightarrow \varphi \qquad \{\varphi\}\ P\ \{\psi\} \qquad \psi \Rightarrow \psi'}{\{\varphi'\}\ P\ \{\psi'\}}$$

Factorial Example

Let's verify the Factorial program using our Hoare rules:

$$\{N \geq 0\}$$
$$i := 0;$$
$$m := 1;$$
$$\{m = i! \wedge N \geq 0\}$$
$$\text{while } i < N \text{ do } \{m = i! \wedge N \geq 0 \wedge i < N\}$$
$$\quad \{m \times (i+1) = (i+1)! \wedge N \geq 0\}$$
$$\quad i := i + 1;$$
$$\quad \{m \times i = i! \wedge N \geq 0\}$$
$$\quad m := m \times i$$
$$\quad \{m = i! \wedge N \geq 0\}$$
$$\text{od } \{m = i! \wedge N \geq 0 \wedge i = N\}$$
$$\{m = N!\}$$

$$\frac{\{\varphi \wedge g\}\ P\ \{\psi\} \quad \{\varphi \wedge \neg g\}\ Q\ \{\psi\}}{\{\varphi\}\ \texttt{if}\ g\ \texttt{then}\ P\ \texttt{else}\ Q\ \texttt{fi}\ \{\psi\}}$$

$$\frac{}{\{\varphi[x := e]\}\ x := e\ \{\varphi\}}$$

$$\frac{\{\varphi \wedge g\}\ P\ \{\varphi\}}{\{\varphi\}\ \texttt{while}\ g\ \texttt{do}\ P\ \texttt{od}\ \{\varphi \wedge \neg g\}}$$

$$\frac{\{\varphi\}\ P\ \{\alpha\} \qquad \{\alpha\}\ Q\ \{\psi\}}{\{\varphi\}\ P; Q\ \{\psi\}}$$

$$\frac{\varphi' \Rightarrow \varphi \qquad \{\varphi\}\ P\ \{\psi\} \qquad \psi \Rightarrow \psi'}{\{\varphi'\}\ P\ \{\psi'\}}$$

note: $(i+1)! = i! \times (i+1)$

Factorial Example

Let's verify the Factorial program using our Hoare rules:

$$\{N \geq 0\}$$
$$i := 0;$$
$$m := 1;\{m = i! \wedge N \geq 0\}$$
$$\{m = i! \wedge N \geq 0\}$$
$$\text{while } i < N \text{ do } \{m = i! \wedge N \geq 0 \wedge i < N\}$$
$$\{m \times (i+1) = (i+1)! \wedge N \geq 0\}$$
$$i := i + 1;$$
$$\{m \times i = i! \wedge N \geq 0\}$$
$$m := m \times i$$
$$\{m = i! \wedge N \geq 0\}$$
$$\text{od } \{m = i! \wedge N \geq 0 \wedge i = N\}$$
$$\{m = N!\}$$

$$\frac{\{\varphi \wedge g\}\ P\ \{\psi\} \quad \{\varphi \wedge \neg g\}\ Q\ \{\psi\}}{\{\varphi\}\ \texttt{if}\ g\ \texttt{then}\ P\ \texttt{else}\ Q\ \texttt{fi}\ \{\psi\}}$$

$$\frac{}{\{\varphi[x := e]\}\ x := e\ \{\varphi\}}$$

$$\frac{\{\varphi \wedge g\}\ P\ \{\varphi\}}{\{\varphi\}\ \texttt{while}\ g\ \texttt{do}\ P\ \texttt{od}\ \{\varphi \wedge \neg g\}}$$

$$\frac{\{\varphi\}\ P\ \{\alpha\} \qquad \{\alpha\}\ Q\ \{\psi\}}{\{\varphi\}\ P; Q\ \{\psi\}}$$

$$\frac{\varphi' \Rightarrow \varphi \qquad \{\varphi\}\ P\ \{\psi\} \qquad \psi \Rightarrow \psi'}{\{\varphi'\}\ P\ \{\psi'\}}$$

note: $(i+1)! = i! \times (i+1)$

Factorial Example

Let's verify the Factorial program using our Hoare rules:

$$\{N \geq 0\}$$
$$i := 0;$$
$$\{1 = i! \wedge N \geq 0\}\ m := 1;\{m = i! \wedge N \geq 0\}$$
$$\{m = i! \wedge N \geq 0\}$$
$$\text{while } i < N \text{ do } \{m = i! \wedge N \geq 0 \wedge i < N\}$$
$$\quad \{m \times (i+1) = (i+1)! \wedge N \geq 0\}$$
$$\quad i := i + 1;$$
$$\quad \{m \times i = i! \wedge N \geq 0\}$$
$$\quad m := m \times i$$
$$\quad \{m = i! \wedge N \geq 0\}$$
$$\text{od } \{m = i! \wedge N \geq 0 \wedge i = N\}$$
$$\{m = N!\}$$

$$\frac{\{\varphi \wedge g\}\ P\ \{\psi\} \quad \{\varphi \wedge \neg g\}\ Q\ \{\psi\}}{\{\varphi\}\ \texttt{if}\ g\ \texttt{then}\ P\ \texttt{else}\ Q\ \texttt{fi}\ \{\psi\}}$$

$$\frac{}{\{\varphi[x := e]\}\ x := e\ \{\varphi\}}$$

$$\frac{\{\varphi \wedge g\}\ P\ \{\varphi\}}{\{\varphi\}\ \texttt{while}\ g\ \texttt{do}\ P\ \texttt{od}\ \{\varphi \wedge \neg g\}}$$

$$\frac{\{\varphi\}\ P\ \{\alpha\} \qquad \{\alpha\}\ Q\ \{\psi\}}{\{\varphi\}\ P; Q\ \{\psi\}}$$

$$\frac{\varphi' \Rightarrow \varphi \qquad \{\varphi\}\ P\ \{\psi\} \qquad \psi \Rightarrow \psi'}{\{\varphi'\}\ P\ \{\psi'\}}$$

note: $(i+1)! = i! \times (i+1)$

Factorial Example

Let's verify the Factorial program using our Hoare rules:

$$\{N \geq 0\}$$
$$i := 0;\{1 = i! \wedge N \geq 0\}$$
$$\{1 = i! \wedge N \geq 0\}\ m := 1;\{m = i! \wedge N \geq 0\}$$
$$\{m = i! \wedge N \geq 0\}$$
$$\text{while } i < N \text{ do } \{m = i! \wedge N \geq 0 \wedge i < N\}$$
$$\quad \{m \times (i+1) = (i+1)! \wedge N \geq 0\}$$
$$\quad i := i + 1;$$
$$\quad \{m \times i = i! \wedge N \geq 0\}$$
$$\quad m := m \times i$$
$$\quad \{m = i! \wedge N \geq 0\}$$
$$\text{od } \{m = i! \wedge N \geq 0 \wedge i = N\}$$
$$\{m = N!\}$$

$$\frac{\{\varphi \wedge g\}\ P\ \{\psi\} \quad \{\varphi \wedge \neg g\}\ Q\ \{\psi\}}{\{\varphi\}\ \texttt{if}\ g\ \texttt{then}\ P\ \texttt{else}\ Q\ \texttt{fi}\ \{\psi\}}$$

$$\frac{}{\{\varphi[x := e]\}\ x := e\ \{\varphi\}}$$

$$\frac{\{\varphi \wedge g\}\ P\ \{\varphi\}}{\{\varphi\}\ \texttt{while}\ g\ \texttt{do}\ P\ \texttt{od}\ \{\varphi \wedge \neg g\}}$$

$$\frac{\{\varphi\}\ P\ \{\alpha\} \qquad \{\alpha\}\ Q\ \{\psi\}}{\{\varphi\}\ P; Q\ \{\psi\}}$$

$$\frac{\varphi' \Rightarrow \varphi \qquad \{\varphi\}\ P\ \{\psi\} \qquad \psi \Rightarrow \psi'}{\{\varphi'\}\ P\ \{\psi'\}}$$

note: $(i+1)! = i! \times (i+1)$

Factorial Example

Let's verify the Factorial program using our Hoare rules:

$$\{N \geq 0\}$$
$$\{1 = 0! \wedge N \geq 0\}\ i := 0;\{1 = i! \wedge N \geq 0\}$$
$$\{1 = i! \wedge N \geq 0\}\ m := 1;\{m = i! \wedge N \geq 0\}$$
$$\{m = i! \wedge N \geq 0\}$$
$$\text{while } i < N \text{ do } \{m = i! \wedge N \geq 0 \wedge i < N\}$$
$$\quad \{m \times (i+1) = (i+1)! \wedge N \geq 0\}$$
$$\quad i := i + 1;$$
$$\quad \{m \times i = i! \wedge N \geq 0\}$$
$$\quad m := m \times i$$
$$\quad \{m = i! \wedge N \geq 0\}$$
$$\text{od } \{m = i! \wedge N \geq 0 \wedge i = N\}$$
$$\{m = N!\}$$

$$\frac{\{\varphi \wedge g\}\ P\ \{\psi\} \quad \{\varphi \wedge \neg g\}\ Q\ \{\psi\}}{\{\varphi\}\ \texttt{if}\ g\ \texttt{then}\ P\ \texttt{else}\ Q\ \texttt{fi}\ \{\psi\}}$$

$$\frac{}{\{\varphi[x := e]\}\ x := e\ \{\varphi\}}$$

$$\frac{\{\varphi \wedge g\}\ P\ \{\varphi\}}{\{\varphi\}\ \texttt{while}\ g\ \texttt{do}\ P\ \texttt{od}\ \{\varphi \wedge \neg g\}}$$

$$\frac{\{\varphi\}\ P\ \{\alpha\} \qquad \{\alpha\}\ Q\ \{\psi\}}{\{\varphi\}\ P; Q\ \{\psi\}}$$

$$\frac{\varphi' \Rightarrow \varphi \qquad \{\varphi\}\ P\ \{\psi\} \qquad \psi \Rightarrow \psi'}{\{\varphi'\}\ P\ \{\psi'\}}$$

note: $(i+1)! = i! \times (i+1)$

Practice Exercise

Example

$m := 1;$
$n := 1;$
$i := 1;$
while $i < N$ do
  $t := m;$
  $m := n;$
  $n := m + t;$
  $i := i + 1$
od

Practice Exercise

Example

$$
\begin{array}{l}
m := 1; \\
n := 1; \\
i := 1; \\
\text{while } i < N \text{ do} \\
\quad t := m; \\
\quad m := n; \\
\quad n := m + t; \\
\quad i := i + 1 \\
\text{od}
\end{array}
$$

- What does this $\mathcal{L}$ program P compute?
- What is a valid Hoare triple $\{\varphi\}P\{\psi\}$ of this program?
- Prove using the inference rules and consequence axiom that this Hoare triple is valid.

Summary

- $\mathcal{L}$: A simple imperative programming language
- Hoare triples (SYNTAX)
- Hoare logic (PROOF)
- Semantics for Hoare logic

Semantics

Nope. That's a topic for another lecture.