A   While language

A.1   Basic language

TODO: add in extra constructs from code TODO: mention other aspects of language, eg variable names TODO: add augmented version used in DCG

$\langle program \rangle \rightarrow \langle statement \rangle \; (; \; \langle program \rangle)^*$

$\langle statement \rangle \rightarrow$ `skip`
$\quad | \quad \langle variable \rangle$ `:=` $\langle aexp \rangle$
$\quad | \quad$ `if` $\langle bexp \rangle$ `then` $\langle statement \rangle$ `else` $\langle statement \rangle$
$\quad | \quad$ `while` $\langle bexp \rangle$ `do` $\langle statement \rangle$
$\quad | \quad$ `(` $\langle statement \rangle$ `)`

$\langle bexp \rangle \rightarrow$ `true` $|$ `false` $|$ `!` $\langle bexp \rangle$ $|$ $\langle bexp \rangle$ `&` $\langle bexp \rangle$
$\quad | \quad \langle aexp \rangle$ `=` $\langle aexp \rangle$
$\quad | \quad \langle aexp \rangle$ `<=` $\langle aexp \rangle$ $|$ $\langle aexp \rangle$ `<` $\langle aexp \rangle$
$\quad | \quad \langle aexp \rangle$ `>=` $\langle aexp \rangle$ $|$ $\langle aexp \rangle$ `>` $\langle aexp \rangle$
$\quad | \quad$ `(` $\langle bexp \rangle$ `)`

$\langle aexp \rangle \rightarrow \langle integer \rangle$ $|$ $\langle variable \rangle$
$\quad | \quad \langle aexp \rangle$ `+` $\langle aexp \rangle$
$\quad | \quad \langle aexp \rangle$ `-` $\langle aexp \rangle$
$\quad | \quad \langle aexp \rangle$ `*` $\langle aexp \rangle$
$\quad | \quad$ `(` $\langle aexp \rangle$ `)`

A.2   Annotations and comments

Comments begun with a `%` symbol and ended with a newline character `\n` may be inserted at any point within a While program. Pre/postconditions and intermediate assertions expressed in an extended language of boolean expressions may also be written within While programs between curly braces, in an extended language of boolean expressions, according to the following rules:

$\langle statement \rangle \rightarrow (\langle assertion \rangle)? \; \langle statement \rangle \; (\langle assertion \rangle)?$

$\langle assertion \rangle \rightarrow \{ \; \langle bexpExtended \rangle \; \}$

$\langle bexpExtended \rangle \rightarrow \langle bexp \rangle$
$\quad | \quad \langle bexp \rangle$ `->` $\langle bexp \rangle$
$\quad | \quad$ `forall` $\langle variable \rangle$ $\langle bexp \rangle$
$\quad | \quad$ `exists` $\langle variable \rangle$ $\langle bexp \rangle$

A.3   Language extensions

- Assigning array elements:

$\langle statement \rangle \rightarrow \langle variable \rangle$ [$\langle aexp \rangle$] := $\langle aexp \rangle$

$\langle aexp \rangle \rightarrow \langle variable \rangle$ [aexp]

- Declaring and calling procedures:

  $\langle statement \rangle \rightarrow$ proc $\langle variable \rangle$ ($\langle variableList \rangle$) ($\langle variableList \rangle$) $\langle statement \rangle$
  $\quad\quad\quad$ | call $\langle variable \rangle$ ($\langle variableList \rangle$) ($\langle aexpList \rangle$)

  Here, $\langle variableList \rangle$ and $\langle aexpList \rangle$ are (possibly empty) comma-separated lists of variable names or expresssions respectively:

  $\langle variableList \rangle \rightarrow \langle variable \rangle$? (, $\langle variable \rangle$)*

  $\langle aexpList \rangle \rightarrow \langle aexp \rangle$? (, $\langle aexp \rangle$)*

  TODO explain different variables given?

- Declaring a block

  $\langle statement \rangle \rightarrow$ begin $\langle variableList \rangle$ $\langle statement \rangle$ end

B Hoare logic

B.1 Basic language

Partial correctness

- Skip axiom $\dfrac{}{\{P\}\ \texttt{skip}\ \{P\}}$

- Assignment axiom $\dfrac{}{\{P[e/x]\}\ x\texttt{:=}e\ \{P\}}$

- Composition rule $\dfrac{\{P\}\ S_1\ \{R\}\ ,\ \{R\}\ S_2\ \{Q\}}{\{P\}\ S_1\texttt{;}S_2\ \{Q\}}$

- Conditional rule $\dfrac{\{b \wedge P\}\ S_1\ \{Q\}\ ,\ \{\neg b \wedge P\}\ S_2\ \{Q\}}{\{P\}\ \texttt{if}\ b\ \texttt{then}\ S_1\ \texttt{else}\ S_2\ \{Q\}}$

- Iteration rule $\dfrac{\{b \wedge I\}\ S\ \{I\}}{\{P\}\ \texttt{while}\ b\ \texttt{do}\ S\ \{\neg b \wedge I\}}$

- Consequence rule 1 $\dfrac{\{P\}\ S\ \{Q\}\ , P^{+} \rightarrow P}{\{P^{+}\}\ S\ \{Q\}}$

- Consequence rule 2 $\dfrac{\{P\}\ S\ \{Q\}\ , Q \rightarrow Q^{-}}{\{P\}\ S\ \{Q^{-}\}}$