Homework assignment

Use the template source file (`hw2.wl` in your repository), implement the semantics and the type system described bellow.

Grammar rules

$$
\begin{aligned}
program &::= CBlock[\{statementSequence\}] \\
statementSequence &::= \epsilon| \\
&\quad statement| \\
&\quad statement, statementSequence \\
statement &::= CDeclare[type, varName]| \\
&\quad expression| \\
&\quad CWhile[expression, \{statementSequence\}] \\
expression &::= number| \\
&\quad varName| \\
&\quad CAssign[varName, expression]| \\
&\quad COperator[unaryOp, expression]| \\
&\quad COperator[binaryOp, \{expression, expression\}] \\
unaryOp &::= Minus| \\
&\quad Not \\
binaryOp &::= Plus| \\
&\quad Subtract| \\
&\quad Times| \\
&\quad Divide| \\
&\quad Greater| \\
&\quad GreaterEqual| \\
&\quad Less| \\
&\quad LessEqual| \\
&\quad Equal| \\
&\quad Unequal| \\
&\quad And| \\
&\quad Or \\
type &::= int| \\
&\quad double
\end{aligned}
\tag{1}
$$

Semantics

Convention: $k, k_1, k_2 \in \{number \cup \{Null\}\}$, $e, e_1, e_2 \in expression$, $stmSeq \in \{statementSequence\}$, $stm* \in statementSequence$ and $\cdot, +$ are standard operators on numbers. Result of logical operations on numbers is 0 if false, 1 if true.

Block rule:

$$\frac{(s, \{stm*\}) \Rightarrow (s', o)}{(s, CBlock[\{stm*\}]) \Rightarrow (s', o)} \tag{2}$$

Statement sequence rules:

$$\frac{}{(s, \{\}) \Rightarrow (s, Null)} \tag{3}$$

$$\frac{(s, stm) \Rightarrow (s', o_1) \quad (s', \{stm*\}) \Rightarrow (s'', o_2)}{(s, \{stm, stm*\}) \Rightarrow (s'', o_2)} \tag{4}$$

Statement rules:

$$\frac{}{(s, CDeclare[type, var]) \Rightarrow (s[var \mapsto Undefined], Null)} \tag{5}$$

$$\frac{(s, e) \Rightarrow (s', k) \wedge k = 0}{(s, While[e, \{stm*\}]) \Rightarrow (s', Null)} \tag{6}$$

$$\frac{(s, e) \Rightarrow (s', k) \wedge k \neq 0 \quad (s', \{stm*\}) \Rightarrow (s'', k_1) \quad (s'', While[e, \{stm*\}]) \Rightarrow (s''', k_2)}{(s, While[e, \{stm*\}]) \Rightarrow (s''', Null)} \tag{7}$$

Expression rules:

$$\frac{}{(s, k) \Rightarrow (s, k)} \tag{8}$$

$$\frac{}{(s, var) \Rightarrow (s, s[var])} \tag{9}$$

$$\frac{(s, e) \Rightarrow (s', k)}{(s, CAssign[var, e]) \Rightarrow (s'[var \mapsto k], k)} \tag{10}$$

Unary operator rules:

$$\frac{(s, e) \Rightarrow (s', k)}{(s, COperator[Minus, e]) \Rightarrow (s', -k)} \tag{11}$$

$$\frac{(s, e) \Rightarrow (s', k) \wedge k = 0}{(s, COperator[Not, e]) \Rightarrow (s', 1)} \tag{12}$$

$$\frac{(s, e) \Rightarrow (s', k) \wedge k \neq 0}{(s, COperator[Not, e]) \Rightarrow (s', 0)} \tag{13}$$

Binary operator rules:

$$\frac{(s, e_1) \Rightarrow (s', k_1) \quad (s', e_2) \Rightarrow (s'', k_2)}{(s, COperator[Plus, \{e1, e2\}]) \Rightarrow (s'', k_1 + k_2)} \tag{14}$$

$$\frac{(s, e_1) \Rightarrow (s', k_1) \quad (s', e_2) \Rightarrow (s'', k_2)}{(s, COperator[Subtract, \{e1, e2\}]) \Rightarrow (s'', k_1 - k_2)} \tag{15}$$

$$\frac{(s, e_1) \Rightarrow (s', k_1) \quad (s', e_2) \Rightarrow (s'', k_2)}{(s, COperator[Times, \{e1, e2\}]) \Rightarrow (s'', k_1 \cdot k_2)} \tag{16}$$

$$\frac{(s, e_1) \Rightarrow (s', k_1) \quad (s', e_2) \Rightarrow (s'', k_2) \wedge k_2 \neq 0}{(s, COperator[Divide, \{e1, e2\}]) \Rightarrow (s'', k_1 \div k_2)} \tag{17}$$

$$\frac{(s, e_1) \Rightarrow (s', k_1) \quad (s', e_2) \Rightarrow (s'', k_2) \wedge k = 0}{(s, COperator[Divide, \{e1, e2\}]) \Rightarrow (s'', \$Failed)} \tag{18}$$

$$\frac{(s, e_1) \Rightarrow (s', k_1) \quad (s', e_2) \Rightarrow (s'', k_2) \wedge k_1 > k_2}{(s, COperator[Greater, \{e1, e2\}]) \Rightarrow (s'', 1)} \tag{19}$$

$$\frac{(s, e_1) \Rightarrow (s', k_1) \quad (s', e_2) \Rightarrow (s'', k_2) \wedge k_1 \leq k_2}{(s, COperator[Greater, \{e1, e2\}]) \Rightarrow (s'', 0)} \tag{20}$$

$$\frac{(s, e_1) \Rightarrow (s', k_1) \quad (s', e_2) \Rightarrow (s'', k_2) \wedge k_1 \geq k_2}{(s, COperator[GreaterEqual, \{e1, e2\}]) \Rightarrow (s'', 1)} \tag{21}$$

$$\frac{(s, e_1) \Rightarrow (s', k_1) \quad (s', e_2) \Rightarrow (s'', k_2) \wedge k_1 < k_2}{(s, COperator[GreaterEqual, \{e1, e2\}]) \Rightarrow (s'', 0)} \tag{22}$$

$$\frac{(s, e_1) \Rightarrow (s', k_1) \quad (s', e_2) \Rightarrow (s'', k_2) \wedge k_1 < k_2}{(s, COperator[Less, \{e1, e2\}]) \Rightarrow (s'', 1)} \tag{23}$$

$$\frac{(s, e_1) \Rightarrow (s', k_1) \quad (s', e_2) \Rightarrow (s'', k_2) \wedge k_1 \geq k_2}{(s, COperator[Less, \{e1, e2\}]) \Rightarrow (s'', 0)} \tag{24}$$

$$\frac{(s, e_1) \Rightarrow (s', k_1) \quad (s', e_2) \Rightarrow (s'', k_2) \wedge k_1 \leq k_2}{(s, COperator[LessEqual, \{e1, e2\}]) \Rightarrow (s'', 1)} \tag{25}$$

$$\frac{(s, e_1) \Rightarrow (s', k_1) \quad (s', e_2) \Rightarrow (s'', k_2) \wedge k_1 > k_2}{(s, COperator[LessEqual, \{e1, e2\}]) \Rightarrow (s'', 0)} \tag{26}$$

$$\frac{(s, e_1) \Rightarrow (s', k_1) \quad (s', e_2) \Rightarrow (s'', k_2) \wedge k_1 = k_2}{(s, COperator[Equal, \{e1, e2\}]) \Rightarrow (s'', 1)} \tag{27}$$

$$\frac{(s, e_1) \Rightarrow (s', k_1) \quad (s', e_2) \Rightarrow (s'', k_2) \wedge k_1 \neq k_2}{(s, COperator[Equal, \{e1, e2\}]) \Rightarrow (s'', 0)} \tag{28}$$

$$\frac{(s, e_1) \Rightarrow (s', k_1) \quad (s', e_2) \Rightarrow (s'', k_2) \wedge k_1 \neq k_2}{(s, COperator[Unequal, \{e1, e2\}]) \Rightarrow (s'', 1)} \tag{29}$$

$$\frac{(s, e_1) \Rightarrow (s', k_1) \quad (s', e_2) \Rightarrow (s'', k_2) \wedge k_1 = k_2}{(s, COperator[Unequal, \{e1, e2\}]) \Rightarrow (s'', 0)} \tag{30}$$

$$\frac{(s, e_1) \Rightarrow (s', k_1) \wedge k_1 = 0}{(s, COperator[And, \{e1, e2\}]) \Rightarrow (s', 0)} \tag{31}$$

$$\frac{(s, e_1) \Rightarrow (s', k_1) \wedge k_1 \neq 0 \quad (s', e_2) \Rightarrow (s'', k_2) \wedge k_2 = 0}{(s, COperator[And, \{e1, e2\}]) \Rightarrow (s'', 0)} \tag{32}$$

$$\frac{(s, e_1) \Rightarrow (s', k_1) \wedge k_1 \neq 0 \quad (s', e_2) \Rightarrow (s'', k_2) \wedge k_2 \neq 0}{(s, COperator[And, \{e1, e2\}]) \Rightarrow (s'', 1)} \tag{33}$$

$$\frac{(s, e_1) \Rightarrow (s', k_1) \wedge k_1 \neq 0}{(s, COperator[Or, \{e1, e2\}]) \Rightarrow (s', 1)} \tag{34}$$

$$\frac{(s, e_1) \Rightarrow (s', k_1) \wedge k_1 = 0 \quad (s', e_2) \Rightarrow (s'', k_2) \wedge k_2 \neq 0}{(s, COperator[Or, \{e1, e2\}]) \Rightarrow (s'', 1)} \tag{35}$$

$$\frac{(s, e_1) \Rightarrow (s', k_1) \wedge k_1 = 0 \quad (s', e_2) \Rightarrow (s'', k_2) \wedge k_2 = 0}{(s, COperator[Or, \{e1, e2\}]) \Rightarrow (s'', 0)} \tag{36}$$

Type rules

Convention: $e, e_1, e_2 \in expression$, $d \in Double$, $n \in Integer$, $var \in varName$, $stmSeq \in \{statementSequence\}$, $stm* \in statementSequence$ and $t \in \{int, double\}$.

$$\frac{}{\Gamma \vdash n : int} \tag{37}$$

$$\frac{}{\Gamma \vdash d : double} \tag{38}$$

$$\frac{\Gamma \vdash \{stm*\} : \diamond}{\Gamma \vdash CBlock[\{stm*\}] : \diamond} \tag{39}$$

$$\frac{}{\Gamma \vdash \{\} : \diamond} \tag{40}$$

$$\frac{\Gamma \vdash e : t \quad \Gamma \vdash \{stm*\} : \diamond}{\Gamma \vdash \{e, stm*\} : \diamond} \tag{41}$$

$$\frac{\Gamma \vdash e : int \quad \Gamma \vdash var : double}{\Gamma \vdash CAssign[var, e] : double} \tag{42}$$

$$\frac{\Gamma \vdash e : t \quad \Gamma \vdash var : t}{\Gamma \vdash CAssign[var, e] : t} \tag{43}$$

$$\frac{\Gamma \vdash CAssign[var, e] : t \quad \Gamma \vdash \{stm*\} : \diamond}{\Gamma \vdash \{CAssign[var, e], stm*\} : \diamond} \tag{44}$$

$$\frac{var \notin dom(\Gamma)}{\Gamma \vdash CDeclare[type, var] : \diamond} \tag{45}$$

$$\frac{\Gamma \vdash CDeclare[type, var] : \diamond \quad \Gamma \cup (var, type) \vdash \{stm*\} : \diamond}{\Gamma \vdash \{CDeclare[type, var], stm*\} : \diamond} \tag{46}$$

$$\frac{\Gamma \vdash e : t \quad stmSeq : \diamond}{\Gamma \vdash CWhile[e, stmSeq] : \diamond} \tag{47}$$

$$\frac{\Gamma \vdash CWhile[e, stmSeq] : \diamond \quad \Gamma \vdash \{stm*\} : \diamond}{\Gamma \vdash \{CWhile[e, stmSeq], stm*\} : \diamond} \tag{48}$$

$$\frac{}{\Gamma \vdash var : \Gamma(var)} \tag{49}$$

$$\frac{\Gamma \vdash var : \Gamma(var) \quad \Gamma \vdash \{stm*\} : \diamond}{\Gamma \vdash \{var, stm*\} : \diamond} \tag{50}$$

$$\frac{\Gamma \vdash e : t}{\Gamma \vdash COperator[unaryOp, e] : t} \tag{51}$$

$$\frac{\Gamma \vdash COperator[unaryOp, e] : t \quad \Gamma \vdash \{stm*\} : \diamond}{\Gamma \vdash \{COperator[unaryOp, e], stm*\} : \diamond} \tag{52}$$

$$\frac{\Gamma \vdash e_1 : int \quad \Gamma \vdash e_2 : double}{\Gamma \vdash COperator[binaryOp, \{e_1, e_2\}] : double} \tag{53}$$

$$\frac{\Gamma \vdash e_1 : double \quad \Gamma \vdash e_2 : int}{\Gamma \vdash COperator[binaryOp, \{e_1, e_2\}] : double} \tag{54}$$

$$\frac{\Gamma \vdash e_1 : t \quad \Gamma \vdash e_2 : t}{\Gamma \vdash COperator[binaryOp, \{e_1, e_2\}] : t} \tag{55}$$

$$\frac{\Gamma \vdash COperator[binaryOp, \{e_1, e_2\}] : t \quad \Gamma \vdash \{stm*\} : \diamond}{\Gamma \vdash \{COperator[binaryOp, \{e_1, e_2\}], stm*\} : \diamond} \tag{56}$$