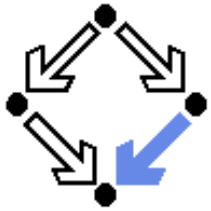
Reasoning about Programs with Interruptions

Wolfgang Schreiner
Wolfgang.Schreiner@risc.uni-linz.ac.at

Research Institute for Symbolic Computation (RISC)
Johannes Kepler University, Linz, Austria
http://www.risc.uni-linz.ac.at

Translating Commands to Formulas



- Judgements:

$$
\begin{aligned}
&C : F \Leftrightarrow \\
&\quad \forall s, s' \in State, e \in Environment : \\
&\qquad executes(control(s)) \wedge [\![C]\!](s, s') \Rightarrow \\
&\qquad\quad [\![F]\!](e)(s, s')
\end{aligned}
$$

- Formulas:

$$
\begin{aligned}
&[\, F \,]^{F_c, F_b, F_r, \{K_1, \ldots, K_m\}}_{I_1, \ldots, I_n} \equiv \\
&\quad (F) \texttt{ AND writesonly } I_1, \ldots, I_n \texttt{ AND} \\
&\quad \texttt{(next.continues => } (F_c)\texttt{) AND} \\
&\quad \texttt{(next.breaks => } (F_b)\texttt{) AND} \\
&\quad \texttt{(next.returns => } (F_r)\texttt{) AND} \\
&\quad \texttt{(next.throws =>} \\
&\qquad \texttt{(next.throws } K_1 \texttt{ OR ... OR next.throws } K_n\texttt{))}
\end{aligned}
$$

We ignore methods for the moment.

Formula Language (Core Syntax)

$F \in$ Formula
$T \in$ Term
$p \in$ Predicate
$f \in$ Function

$$
\begin{aligned}
F ::= & \texttt{TRUE} \mid \texttt{FALSE} \\
& \mid p(T_1, \ldots, T_n) \mid T_1 \texttt{ = } T_2 \mid T_1 \texttt{ /= } T_2 \\
& \mid \texttt{readsonly} \mid \texttt{writesonly } I_1, \ldots, I_n \\
& \mid \texttt{!}F \mid F_1 \texttt{ AND } F_2 \mid F_1 \texttt{ OR } F_2 \mid F_1 \texttt{ => } F_2 \mid F_1 \texttt{ <=> } F_2 \\
& \mid F_1 \texttt{ XOR } F_2 \mid \texttt{IF } F \texttt{ THEN } F_1 \texttt{ ELSE } F_2 \mid \\
& \mid \texttt{FORALL \$}I_1, \ldots, \texttt{\$}I_n\texttt{: } F \mid \texttt{EXISTS \$}I_1, \ldots, \texttt{\$}I_n\texttt{: } F \\
& \mid \texttt{LET \$}I_1\texttt{=}T_1, \ldots, \texttt{\$}I_n\texttt{=}T_n \texttt{ IN } F \\
T ::= & I \mid I\texttt{'} \mid \texttt{\$}I \mid f(T_1, \ldots, T_n) \\
& \mid \texttt{IF } F \texttt{ THEN } T_1 \texttt{ ELSE } T_2 \\
& \mid \texttt{LET \$}I_1\texttt{=}T_1, \ldots, \texttt{\$}I_n\texttt{=}T_n \texttt{ IN } T \mid \texttt{SUCH \$}I\texttt{: } F
\end{aligned}
$$

Formula Language (Core Semantics)

$[\![\sqcup]\!] : \textbf{Formula} \rightarrow \textbf{Environment} \rightarrow \textbf{StateRelation}$

$[\![\texttt{TRUE}]\!](e)(s, s') \;\Leftrightarrow\; \text{TRUE}$

$[\![\texttt{FALSE}]\!](e)(s, s') \;\Leftrightarrow\; \text{FALSE}$

$[\![p\texttt{(}T_1, \ldots, T_n\texttt{)}]\!](e)(s, s') \Leftrightarrow$

$\quad [\![p]\!]([\![T_1]\!](e)(s, s'), \ldots, [\![T_n]\!](e)(s, s'))$

$[\![T_1 \texttt{ = } T_2]\!](e)(s, s') \Leftrightarrow [\![T_1]\!](e)(s, s') = [\![T_2]\!](e)(s, s')$

$[\![\texttt{readsonly}]\!](e)(s, s') \Leftrightarrow s = s'$

$[\![\texttt{writesonly } l_1, \ldots, l_n]\!](e)(s, s') \Leftrightarrow s = s' \text{ EXCEPT } l_1, \ldots, l_n$

$[\![\texttt{!}F]\!](e)(s, s') \Leftrightarrow \neg[\![F]\!](e)(s, s')$

$[\![F_1 \texttt{ AND } F_2]\!](e)(s, s') \Leftrightarrow [\![F_1]\!](e)(s, s') \wedge [\![F_2]\!](e)(s, s')$

$\ldots$

$[\![\texttt{FORALL \$}l_1, \ldots, \texttt{\$}l_n\texttt{:}\ F]\!](e)(s, s') \Leftrightarrow$

$\quad \forall v_1, \ldots, v_n \in \mathit{Value} : [\![F]\!](e[\, l_1 \mapsto v_1, \ldots, l_n \mapsto v_n \,])(s, s')$

$[\![\texttt{EXISTS \$}l_1, \ldots, \texttt{\$}l_n\texttt{:}\ F]\!](e)(s, s') \Leftrightarrow$

$\quad \exists v_1, \ldots, v_n \in \mathit{Value} : [\![F]\!](e[\, l_1 \mapsto v_1, \ldots, l_n \mapsto v_n \,])(s, s')$

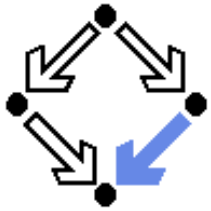
$\ldots$

Formula Language (Core Semantics)

$[\![\sqcup]\!] : \textbf{Term} \rightarrow \textbf{Environment} \rightarrow \textbf{BinaryStateFunction}$

$[\![I]\!](e)(s, s') = read(s, I)$
$[\![I\text{'}]\!](e)(s, s') = read(s', I)$
$[\![\$I]\!](e)(s, s') = e(I)$
$[\![f(T_1, \ldots, T_n)]\!](e)(s, s') =$
$\quad [\![f]\!]([\![T_1]\!](e)(s, s'), \ldots, [\![T_n]\!](e)(s, s'))$
$\ldots$

Formula Language: Extended Syntax



$S \in \text{State}$

$$
\begin{aligned}
F ::= \ & \ldots \\
& \mid \texttt{ALLSTATE}\ \#l_1, \ldots, \#l_n\texttt{:}\ F \\
& \mid \texttt{EXSTATE}\ \#l_1, \ldots, \#l_n\texttt{:}\ F \\
& \mid S_1 \texttt{ == } S_2 \\
& \mid S\texttt{.executes} \mid S\texttt{.continues} \mid S\texttt{.breaks} \\
& \mid S\texttt{.returns} \mid S\texttt{.throws} \mid S\texttt{.throws}\ l
\end{aligned}
$$

$$T ::= \ldots \mid S\texttt{.value}$$

$$S ::= \texttt{now} \mid \texttt{next} \mid \#l$$

Formula Language: Extended Semantics

$[\![\,\_\,]\!] : \mathbf{Formula} \rightarrow \textbf{Environment} \rightarrow \textbf{StateRelation}$

$\ldots$

$[\![\texttt{ALLSTATE}\ \#l_1, \ldots, \#l_n\texttt{:}\ F]\!](e)(s, s') \Leftrightarrow$
$\quad \forall c_1, \ldots, c_n \in \mathit{Control} : [\![F]\!](e[\, l_1 \mapsto c_1, \ldots, l_n \mapsto c_n \,]_c)(s, s')$

$[\![\texttt{EXSTATE}\ \#l_1, \ldots, \#l_n\texttt{:}\ F]\!](e)(s, s') \Leftrightarrow$
$\quad \exists c_1, \ldots, c_n \in \mathit{Control} : [\![F]\!](e[\, l_1 \mapsto c_1, \ldots, l_n \mapsto c_n \,]_c)(s, s')$

$[\![S_1 \texttt{ == } S_2]\!](e)(s, s') \Leftrightarrow [\![S_1]\!](e)(s, s') = [\![S_2]\!](e)(s, s')$

$[\![S\texttt{.executes}]\!](e)(s, s') \Leftrightarrow \mathit{executes}([\![S]\!](e)(s, s'))$

$[\![S\texttt{.continues}]\!](e)(s, s') \Leftrightarrow \mathit{continues}([\![S]\!](e)(s, s'))$

$[\![S\texttt{.breaks}]\!](e)(s, s') \Leftrightarrow \mathit{breaks}([\![S]\!](e)(s, s'))$

$[\![S\texttt{.returns}]\!](e)(s, s') \Leftrightarrow \mathit{returns}([\![S]\!](e)(s, s'))$

$[\![S\texttt{.throws}]\!](e)(s, s') \Leftrightarrow \mathit{throws}([\![S]\!](e)(s, s'))$

$[\![S\texttt{.throws}\ I]\!](e)(s, s') \Leftrightarrow$
$\quad \textsc{let}\ c = [\![S]\!](e)(s, s')\ \textsc{in}\ \mathit{throws}(c) \wedge \mathit{key}(c) = I$

Formula Language: Extended Semantics

$[\![\sqcup]\!] : \mathbf{Term} \rightarrow \textbf{Environment} \rightarrow (\textbf{State} \times \textbf{State}) \rightarrow \textbf{Value}$

$\ldots$

$[\![S\,.\,\mathtt{value}]\!](e)(s, s') = \mathit{value}([\![S]\!](e)(s, s'))$

$[\![\sqcup]\!] : \mathbf{State} \rightarrow \textbf{Environment} \rightarrow (\textbf{State} \times \textbf{State}) \rightarrow \textbf{Control}$

$[\![\mathtt{now}]\!](e)(s, s') = \mathit{control}(s)$

$[\![\mathtt{next}]\!](e)(s, s') = \mathit{control}(s')$

$[\![\#l]\!](e)(s, s') = e(l)_c$

Generic Rules

$$\frac{C : [\,F\,]^{F_c,F_b,F_r,\{K_1,\ldots,K_m\}}_{I_1,\ldots,I_n} \qquad \mathsf{p} \text{ is a permutation of } \{1,\ldots,n\}}{C : [\,F\,]^{F_c,F_b,F_r,\{K_1,\ldots,K_m\}}_{I_{p(1)},\ldots,I_{p(n)}}}$$

$$\frac{C : [\,F\,]^{F_c,F_b,F_r,\{K_1,\ldots,K_m\}}_{I_1,\ldots,I_n} \qquad I \neq I_1 \wedge \ldots \wedge I \neq I_n}{C : [\,F \texttt{ AND } I\texttt{'}= I\,]^{F_c,F_b,F_r,\{K_1,\ldots,K_m\}}_{I_1,\ldots,I_n,I}}$$

$$\frac{C : [\,F\,]^{\texttt{FALSE},\texttt{FALSE},\texttt{FALSE},\emptyset}_{I_1,\ldots,I_n}}{C : [\,F\,]_{I_1,\ldots,I_n}}$$

$$\frac{C \text{ is a program without interruptions} \qquad C : [\,F\,]_{I_1,\ldots,I_n}}{C : [\,F\,]^{\texttt{FALSE},\texttt{FALSE},\texttt{FALSE},\emptyset}_{I_1,\ldots,I_n}}$$

Assignments and Declarations

$$\frac{E \simeq T}{I \texttt{ = } E : [\, I\texttt{'= } T \texttt{ AND next.executes} \,]_{I}^{\texttt{FALSE,FALSE,FALSE,}\emptyset}}$$

$$\frac{\begin{array}{l} C : [\, F \,]_{I_1,\ldots,I_n,I}^{F_c,F_b,F_r,\{K_1,\ldots,K_m\}} \\ I_a \neq I_b \end{array}}{\begin{array}{l} \texttt{var } I\texttt{; } C : \\ \quad [\texttt{EXISTS \$}I_a\texttt{,\$}I_b : F[\texttt{\$}I_a/I, \texttt{\$}I_b/I\texttt{'}]]_{I_1,\ldots,I_n}^{F_c,F_b,F_r,\{K_1,\ldots,K_m\}} \end{array}}$$

$$\frac{\begin{array}{l} C : [\, F \,]_{I_1,\ldots,I_n,I}^{F_c,F_b,F_r,\{K_1,\ldots,K_m,\}} \\ I_a \neq I_b \\ E \simeq T \end{array}}{\begin{array}{l} \texttt{var } I\texttt{=}E\texttt{; } C : \\ \quad [\texttt{EXISTS \$}I_a\texttt{,\$}I_b : \\ \qquad \texttt{\$}I_a\texttt{=}T \texttt{ AND } F[\texttt{\$}I_a/I, \texttt{\$}I_b/I\texttt{'}]]_{I_1,\ldots,I_n}^{F_c,F_b,F_r,\{K_1,\ldots,K_m\}} \end{array}}$$

Command Sequences (w/o Interruption)

$$\frac{C_1 : [\,F_1\,]^{\texttt{FALSE},\texttt{FALSE},\texttt{FALSE},\emptyset}_{I_1,\ldots,I_n} \qquad C_2 : [\,F_2\,]^{F_c,F_b,F_r,\{K_1,\ldots,K_m\}}_{I_1,\ldots,I_n}}{\begin{array}{l} C_1\,;C_2 : \\ \quad [\texttt{EXISTS}\ \$I_1,\ldots,\$I_n\texttt{:}\ \texttt{EXSTATE}\ \#I_s\texttt{:} \\ \qquad F_1[\#I_s/\texttt{next}][\$I_1/I_1\texttt{'},\ldots,\$I_n/I_n\texttt{'}]\ \texttt{AND} \\ \qquad F_2[\#I_s/\texttt{now}][\$I_1/I_1,\ldots,\$I_n/I_n]\,]^{F_c,F_b,F_r,\{K_1,\ldots,K_m\}}_{I_1,\ldots,I_n} \end{array}}$$

Command Sequences (with Interruption)

$$\frac{C_1 : [\,F_1\,]^{F_{c1},F_{b1},F_{r1},\{K_1,\ldots,K_m\}}_{I_1,\ldots,I_n} \qquad C_2 : [\,F_2\,]^{F_{c1},F_{b1},F_{r1},\{L_1,\ldots,L_o\}}_{I_1,\ldots,I_n}}{C_1\,;C_2 :}$$

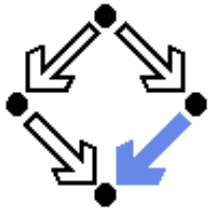
$$\begin{array}{l}
[\,\texttt{EXISTS}\ \$I_1,\ldots,\$I_n\texttt{:}\ \texttt{EXSTATE}\ \#I_s\texttt{:} \\
\quad F_1[\#I_s/\texttt{next}][\$I_1/I_1\texttt{'},\ldots,\$I_n/I_n\texttt{'}]\ \texttt{AND} \\
\quad \texttt{IF}\ \#I_s\texttt{.executes THEN} \\
\qquad F_2[\#I_s/\texttt{now}][\$I_1/I_1,\ldots,\$I_n/I_n] \\
\quad \texttt{ELSE} \\
\qquad I_1\texttt{'=}\$I_1\ \texttt{AND}\ \ldots\ \texttt{AND}\ I_n\texttt{'=}\$I_n\ \texttt{AND}\ \texttt{next==}\#I_s \\
]^{F_{c1}\ \texttt{OR}\ F_{c2},F_{b1}\ \texttt{OR}\ F_{b2},F_{r1}\ \texttt{OR}\ F_{r2},\{K_1,\ldots,K_m,L_1,\ldots,L_o\}}_{I_1,\ldots,I_n}
\end{array}$$

Conditionals

$$\frac{C : [\,F\,]^{F_c,F_b,F_r,\{K_1,\ldots,K_m\}}_{I_1,\ldots,I_n} \quad E \simeq F_0}{\texttt{if (}E\texttt{) } C : [\,\texttt{IF } F_0 \texttt{ THEN } F \texttt{ ELSE readsonly}\,]^{F_c,F_b,F_r,\{K_1,\ldots,K_m\}}_{I_1,\ldots,I_n}}$$

$$\frac{C_1 : [\,F_1\,]^{F_{c1},F_{b1},F_{r1},\{K_1,\ldots,K_m\}}_{I_1,\ldots,I_n} \quad C_2 : [\,F_2\,]^{F_{c2},F_{b2},F_{r2},\{L_1,\ldots,L_o\}}_{I_1,\ldots,I_n} \quad E \simeq F_0}{\begin{array}{l}\texttt{if (}E\texttt{) } C_1 \texttt{ else } C_2 : \\ \quad [\,\texttt{IF } F_0 \texttt{ THEN } F_1 \\ \qquad \texttt{ELSE } F_2\,]^{F_{c1}\texttt{ OR }F_{c2},F_{b1}\texttt{ OR }F_{b2},F_{r1}\texttt{ OR }F_{r2},\{K_1,\ldots,K_m,L_1,\ldots,L_o\}}_{I_1,\ldots,I_n}\end{array}}$$

Interruptions



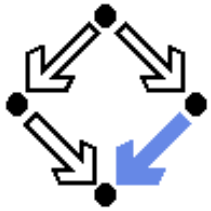
$$\texttt{continue} : [\texttt{next.continues}]_{\sqcup}^{\texttt{TRUE},\texttt{FALSE},\texttt{FALSE},\emptyset}$$

$$\texttt{break} : [\texttt{next.breaks}]_{\sqcup}^{\texttt{FALSE},\texttt{TRUE},\texttt{FALSE},\emptyset}$$

$$\frac{T \simeq E}{\begin{array}{l}\texttt{return}\ E : \\ \quad [\texttt{next.returns} \\ \quad \texttt{AND next.value=}T\,]_{\sqcup}^{\texttt{FALSE},\texttt{FALSE},\texttt{TRUE},\emptyset}\end{array}}$$

$$\frac{T \simeq E}{\begin{array}{l}\texttt{throw}\ I\ E : \\ \quad [\texttt{next.throws}\ I \\ \quad \texttt{AND next.value=}T\,]_{\sqcup}^{\texttt{FALSE},\texttt{FALSE},\texttt{FALSE},\{I\}}\end{array}}$$

Interruption Handlers



$$
\frac{
\begin{array}{l}
C_1 : [\, F_1 \,]^{F_{c1},F_{b1},F_{r1},\{K_1,\ldots,K_m\}}_{I_1,\ldots,I_n} \\
C_2 : [\, F_2 \,]^{F_{c2},F_{b2},F_{r2},\{L_1,\ldots,L_o\}}_{I_1,\ldots,I_n} \\
I_a \neq I_b \\
\{I_a, I_b\} \cap \{I_1, \ldots, I_n\} = \emptyset
\end{array}
}{
\begin{array}{l}
\texttt{try } C_1 \texttt{ catch(}I_k\ I_v\texttt{) } C_2 : \\
\quad [\texttt{EXISTS } \$I_1,\ldots,\$I_n\texttt{: EXSTATE } \#I_s\texttt{:} \\
\qquad F_1[\#I_s/\texttt{next}][\$I_1/I_1\texttt{'},\ldots,\$I_n/I_n\texttt{'}] \texttt{ AND} \\
\qquad \texttt{IF } \#I_s\texttt{.throws } I_k \texttt{ THEN} \\
\qquad\quad \texttt{EXISTS } \$I_a,\$I_b\texttt{: EXSTATE } \#I_t\texttt{:} \\
\qquad\qquad \$I_a \texttt{ = } \#I_s\texttt{.value AND } \#I_t\texttt{.executes AND} \\
\qquad\qquad F_2[\#I_t/\texttt{now}][\$I_a/I_v][\$I_1/I_1,\ldots,\$I_n/I_n][\$I_b/I_v\texttt{'}] \\
\qquad \texttt{ELSE} \\
\qquad\quad I_1\texttt{'=}\$I_1 \texttt{ AND ... AND } I_n\texttt{'=}\$I_n \texttt{ AND next==}\#I_s \\
\quad]^{F_{c1} \texttt{ OR } F_{c2},F_{b1} \texttt{ OR } F_{b2},F_{r1} \texttt{ OR } F_{r2},(\{K_1,\ldots,K_m\}\setminus\{I_k\})\cup\{L_1,\ldots,L_o\}}_{I_1,\ldots,I_n}
\end{array}
}
$$

Loops w/o Interruptions

$$
\begin{aligned}
&\mathit{finiteExecution}: \\
&\quad \mathbb{P}(\mathbb{N} \times \mathit{State}^{\infty} \times \mathit{State} \times \mathit{StateFunction} \times \mathit{StateRelation}) \\
&\mathit{finiteExecution}(k, t, s, E, C) \Leftrightarrow \\
&\quad t(0) = s \wedge \forall i \in \mathbb{N}_k : E(t(i)) = \text{TRUE} \wedge C(t(i), t(i+1))
\end{aligned}
$$

$$
\begin{aligned}
&[\![\, \texttt{while (}E\texttt{) } C \,]\!](s, s') \Leftrightarrow \\
&\quad \exists k \in \mathbb{N}, t \in \mathit{State}^{\infty} : \\
&\qquad \mathit{finiteExecution}(k, t, s, [\![\, E \,]\!], [\![\, C \,]\!]) \wedge \\
&\qquad [\![\, E \,]\!](t(k)) \neq \text{TRUE} \wedge t(k) = s'
\end{aligned}
$$

$$
\frac{\begin{array}{l} C : [\, F \,]_{I_1,\ldots,I_n} \\ E \simeq H \end{array}}{\texttt{while (}E\texttt{) } C : [\, !H[I_1\texttt{'}/I_1, \ldots, I_n\texttt{'}/I_n] \,]_{I_1,\ldots,I_n}}
$$

Loops w/o Interruptions with Invariants

$$
\begin{array}{l}
\mathit{Invariant}(G, H, F)_{I_1,\ldots,I_n} \equiv \\
\quad G \text{ has no free variables} \wedge \\
\quad \forall e \in \mathit{Environment}, s, s' \in \mathit{Store} : \\
\qquad [\![\texttt{FORALL } \$I_1, \ldots, \$I_n : \\
\qquad\quad (G[\$I_1/I_1\texttt{'}, \ldots, \$I_n/I_n\texttt{'}] \texttt{ AND } H[\$I_1/I_1, \ldots, \$I_n/I_n] \\
\qquad\qquad \texttt{AND } F[\$I_1/I_1, \ldots, \$I_n/I_n]) \texttt{ => } G)]\!](e)(s, s')
\end{array}
$$

$$
\frac{
\begin{array}{l}
C : [\,F\,]_{I_1,\ldots,I_n} \\
E \simeq H \\
\mathit{Invariant}(G, H, F)_{I_1,\ldots,I_n}
\end{array}
}{
\begin{array}{l}
\texttt{while } (E)\ C : [\,!H[I_1\texttt{'}/I_1, \ldots, I_n\texttt{'}/I_n] \texttt{ AND} \\
\quad (G[I_1/I_1\texttt{'}, \ldots, I_n/I_n\texttt{'}] \texttt{ => } G)\,]_{I_1,\ldots,I_n}
\end{array}
}
$$

Loops with Interuptions

$$
\begin{aligned}
&\mathit{finiteExecution} : \\
&\mathbb{P}(\mathbb{N} \times \mathit{State}^\infty \times \mathit{State}^\infty \times \mathit{State} \times \mathit{StateFunction} \times \mathit{StateRelation}) \\
&\mathit{finiteExecution}(k, t, u, s, E, C) \Leftrightarrow \\
&\quad t(0) = s \wedge u(0) = s \wedge \\
&\quad \forall i \in \mathbb{N}_k : \\
&\qquad \neg \mathit{breaks}(\mathit{control}(u(i))) \wedge \mathit{executes}(\mathit{control}(t(i))) \wedge \\
&\qquad E(t(i)) = \text{TRUE} \wedge C(t(i), u(i+1)) \wedge \\
&\qquad \text{IF } \mathit{continues}(\mathit{control}(u(i+1))) \vee \mathit{breaks}(\mathit{control}(u(i+1))) \\
&\qquad\quad \text{THEN } t(i+1) = \mathit{execute}(u(i+1)) \\
&\qquad\quad \text{ELSE } t(i+1) = u(i+1) \\
&[\![\texttt{while (}E\texttt{) } C]\!](s, s') \Leftrightarrow \\
&\quad \exists k \in \mathbb{N}, t, u \in \mathit{State}^\infty : \\
&\qquad \mathit{finiteExecution}(k, t, u, s, [\![E]\!], [\![C]\!]) \wedge \\
&\qquad ([\![E]\!](t(k)) \neq \text{TRUE} \vee \\
&\qquad\quad \neg(\mathit{executes}(\mathit{control}(u(k))) \vee \\
&\qquad\qquad \mathit{continues}(\mathit{control}(u(k))))) \wedge \\
&\qquad t(k) = s'
\end{aligned}
$$

Loops with Interruptions

$$\frac{C : [\,F\,]^{F_c,F_b,F_r;K_1,\ldots,K_m}_{I_1,\ldots,I_n}}{\begin{array}{l}\texttt{while}\ (E)\ C:\\ \quad [\,\texttt{!next.continues AND}\\ \qquad \texttt{!next.breaks}\,]^{\texttt{FALSE},\texttt{FALSE},F_r;K_1,\ldots,K_m}_{I_1,\ldots,I_n}\end{array}}$$

$$\frac{\begin{array}{l}C : [\,F\,]^{F_c,\texttt{FALSE},F_r;K_1,\ldots,K_m}_{I_1,\ldots,I_n}\\ E \simeq H\end{array}}{\begin{array}{l}\texttt{while}\ (E)\ C:\\ \quad [\,\texttt{!next.continues AND !next.breaks AND}\\ \quad (\texttt{next.executes =>}\\ \qquad \texttt{!}H[\texttt{next}/\texttt{now}][I_1\texttt{'}/I_1,\ldots,I_n\texttt{'}/I_n])\\ \quad]^{\texttt{FALSE},\texttt{FALSE},F_r;K_1,\ldots,K_m}_{I_1,\ldots,I_n}\end{array}}$$

Loops with Interruptions and Invariants

$$
\begin{array}{l}
\mathit{Invariant}(G, H, F)_{I_1,\ldots,I_n} \equiv \\
\quad G \text{ has no free (mathematical or state) variables} \wedge \\
\quad \forall e \in \mathit{ValueEnv}, s, s' \in \mathit{Store} : \\
\quad\quad [\![\texttt{FORALL } \$I_1, \ldots, \$I_n \texttt{: ALLSTATE } \#I_s \texttt{,} \#I_t \texttt{:} \\
\quad\quad\quad (G[\#I_s/\texttt{next}][\$I_1/I_1\texttt{'}, \ldots, \$I_n/I_n\texttt{'}] \\
\quad\quad\quad\quad \texttt{AND } (\#I_s\texttt{.executes OR } \#I_s\texttt{.continues}) \\
\quad\quad\quad\quad \texttt{AND } \#I_t\texttt{.executes} \\
\quad\quad\quad\quad \texttt{AND } H[\#I_t/\texttt{now}][\$I_1/I_1, \ldots, \$I_n/I_n] \\
\quad\quad\quad\quad \texttt{AND } F[\#I_t/\texttt{now}][\$I_1/I_1, \ldots, \$I_n/I_n]) \\
\quad\quad\quad \texttt{=>} \ G)]\!](e)(s, s')
\end{array}
$$

Loops with Interruptions and Invariants

$$\frac{C : [\,F\,]^{F_c,F_b,F_r;K_1,\ldots,K_m}_{I_1,\ldots,I_n} \qquad Invariant(G,H,F)_{I_1,\ldots,I_n}}{\begin{array}{l}\texttt{while}\ (E)\ C:\\ \quad [\,\texttt{!next.continues AND !next.breaks AND}\\ \quad (G[\texttt{now}/\texttt{next}][I_1/I_1\texttt{'},\ldots,I_n/I_n\texttt{'}]\ \texttt{=>}\\ \qquad \texttt{EXISTS}\ \#I_s : G[\#I_s/\texttt{next}]\ \texttt{AND}\\ \qquad\quad \texttt{IF}\ \#I_s\texttt{.continues OR}\ \#I_s\texttt{.breaks}\\ \qquad\qquad \texttt{THEN next.executes}\\ \qquad\qquad \texttt{ELSE next == }\#I_s)\\ \quad]^{\texttt{FALSE},\texttt{FALSE},F_r;K_1,\ldots,K_m}_{I_1,\ldots,I_n}\end{array}}$$

Loops with Interruptions and Invariants

$$
\frac{
\begin{array}{l}
C : [\,F\,]^{F_c,\texttt{FALSE},F_r;K_1,\ldots,K_m}_{I_1,\ldots,I_n} \\
E \simeq H \\
\mathit{Invariant}(G,H,F)_{I_1,\ldots,I_n}
\end{array}
}{
\begin{array}{l}
\texttt{while}\ (E)\ C : \\
\quad [\,\texttt{!next.continues AND !next.breaks AND} \\
\quad (\texttt{next.executes =>} \\
\qquad \texttt{!}H[\texttt{next/now}][I_1\texttt{'}/I_1,\ldots,I_n\texttt{'}/I_n])\ \texttt{AND} \\
\quad (G[\texttt{now/next}][I_1/I_1\texttt{'},\ldots,I_n/I_n\texttt{'}]\ \texttt{=>} \\
\qquad \texttt{EXISTS}\ \#I_s : G[\#I_s/\texttt{next}]\ \texttt{AND} \\
\qquad\quad \texttt{IF}\ \#I_s\texttt{.continues OR}\ \#I_s\texttt{.breaks} \\
\qquad\qquad \texttt{THEN next.executes} \\
\qquad\qquad \texttt{ELSE next ==}\ \#I_s) \\
\quad]^{\texttt{FALSE},\texttt{FALSE},F_r;K_1,\ldots,K_m}_{I_1,\ldots,I_n}
\end{array}
}
$$

Outlook

- Further calculi:
  - $C \downarrow F$
  - $\text{PRE}(C, Q) = P$
  - $\text{POST}(C, P) = Q$
  - $\text{TRANS}(C, P) = C'$
- Expressions and interruptions

  `X = 1/0`
- Methods

  $I_r = I_m(E_1, \ldots, E_p)$

Stay tuned for more to come in this seminar.