Alcino Cunha

SPECIFICATION AND MODELING

FIRST-ORDER LINEAR TEMPORAL LOGIC

Universidade do Minho & INESC TEC

2019/20

TRASH

TRASH

Design a trash component such that:

- A deleted file can still be restored if the trash is not emptied

TRASH BEHAVIOUR

```
var sig File {}
var sig Trash in File {}

pred delete[f : File] { ... }
pred restore[f : File] { ... }
pred empty { ... }
pred do_nothing { ... }

fact {
  no Trash
  always (
    (some f: File | delete[f] or restore[f]) or empty or do_nothing
  )
}
```

SOME TRASH ASSERTIONS

```
assert restoreAfterDelete {
  -- Every restored file was once deleted
  always (all f : File | restore[f] implies once delete[f])
}

assert deleteAll {
  -- If the trash contains all files and is emptied
  -- then no files will ever exist afterwards
  always ((File in Trash and empty) implies after (always no File))
}
```

FIRST-ORDER LINEAR TEMPORAL LOGIC

FIRST-ORDER LINEAR TEMPORAL LOGIC

- Electrum includes temporal connectives from *Linear Temporal Logic* (LTL)
  - Both future and past operators
- An LTL formula is interpreted in a state of a *trace* (infinite sequence of states)
  - A formula is valid in a trace iff it is valid in its initial state
  - A formula is valid in a system iff it is valid in all possible traces

FUTURE OPERATORS

Electrum	Math		Meaning
always ϕ	$\mathsf{G}\,\phi$	$\square\,\phi$	ϕ is always true from now on
eventually ϕ	$\mathsf{F}\,\phi$	$\diamond\,\phi$	ϕ will eventually be true
after ϕ	$\mathsf{X}\,\phi$	$\bigcirc\,\phi$	ϕ will be true in the next state
ϕ **until** ψ	$\phi\ \mathsf{U}\ \psi$		ψ will eventually be true and ϕ is true until then
ϕ **releases** ψ	$\phi\ \mathsf{R}\ \psi$		ψ can only be false after ϕ is true

PAST OPERATORS

Electrum	Math	Meaning
historically ϕ	$\mathsf{H}\,\phi$	ϕ was always true
once ϕ	$\mathsf{O}\,\phi$	ϕ was once true
before ϕ	$\mathsf{Y}\,\phi$	ϕ was true in the previous state
ϕ **since** ψ	$\phi\ \mathsf{S}\ \psi$	ψ was once true and ϕ has been true afterwards
ϕ **triggered** ψ	$\phi\ \mathsf{T}\ \psi$	if ϕ was once true, then ψ has been true onwards

SEMANTICS BY EXAMPLE

```
var lone sig A {}
var lone sig B {}
```

SEMANTICS BY EXAMPLE

eventually some B

after (**some** A **and some** B)

some B **releases some** A

once some A

always (**some** B **implies eventually some** A)

eventually always some A

always eventually some A

SEMANTICS BY EXAMPLE


```
not always some A

not before some B

not always (some A implies eventually some B)

not eventually always some B

not always eventually some B
```

SEMANTICS BY EXAMPLE

always some A

once some B

not eventually some B

SEMANTICS BY EXAMPLE

before some B

some B **until some** A

not historically some A

TRASH

THE DESIRED TRASH ASSERTION

```
pred restoreEnabled[f : File] {
    f in Trash
}

assert restoreIsPossibleBeforeEmpty {
  --  a deleted file can still be restored if the trash is not emptied
  always (all f:File | delete[f] implies
            (empty releases restoreEnabled[f]))
}
```


DEMO

THE DESIRED TRASH ASSERTION

```
pred restoreEnabled[f : File] {
    f in Trash
}

assert restoreIsPossibleBeforeEmpty {
  --  a deleted file can still be restored if the trash is not emptied
  always (all f:File | delete[f] implies
           after ((empty or restore[f]) releases restoreEnabled[f]))
}
```

FIRST-ORDER LINEAR TEMPORAL LOGIC

SYNTAX

$$
\begin{array}{rcl}
\phi & ::= & \mathsf{G}\,\phi \\
 & | & \mathsf{F}\,\phi \\
 & | & \mathsf{X}\,\phi \\
 & | & \phi\ \mathsf{U}\ \psi \\
 & | & \phi\ \mathsf{R}\ \psi \\
 & | & \mathsf{H}\,\phi \\
 & | & \mathsf{O}\,\phi \\
 & | & \mathsf{Y}\,\phi \\
 & | & \phi\ \mathsf{S}\ \psi \\
 & | & \phi\ \mathsf{T}\ \psi \\
 & | & \ldots
\end{array}
$$

$$
\begin{array}{rcl}
\Phi & ::= & \Phi' \\
 & | & \ldots
\end{array}
$$

FIRST-ORDER TEMPORAL STRUCTURES

- The semantics of a first-order *temporal* formula is defined over a first-order *temporal* structure (aka *model*) $\mathcal{M} = (\mathcal{D}, \pi)$
  - $\mathcal{D}$ is a non-empty domain of interpretation (or discourse) with equality
  - π is an infinite sequence of possible interpretations of the predicates (a trace)
  - Given $i \in \mathbb{N}$, we have $\pi(i)(P) \subseteq \mathcal{D}^{\mathrm{ar}(P)}$
- For interpreting (free) variables we still need an assignment $\mathcal{A}$
- The fact that a formula ϕ is valid in the i-th state of a model $\mathcal{M}$ with assignment $\mathcal{A}$ is denoted by $\mathcal{M}, \mathcal{A}, i \models \phi$
- A formula ϕ is valid in a model $\mathcal{M}$ with assignment $\mathcal{A}$, denoted by $\mathcal{M}, \mathcal{A}, i \models \phi$, iff $\mathcal{M}, \mathcal{A}, 0 \models \phi$
- If the formula is closed we write just $\mathcal{M} \models \phi$, assuming $\mathcal{A}$ to be the empty assignment

SEMANTICS

$$
\begin{array}{rcl}
\mathcal{M}, \mathcal{A}, i \models \mathsf{G}\,\phi & \text{iff} & \forall j \geq i \,.\, \mathcal{M}, \mathcal{A}, j \models \phi \\
\mathcal{M}, \mathcal{A}, i \models \mathsf{F}\,\phi & \text{iff} & \exists j \geq i \,.\, \mathcal{M}, \mathcal{A}, j \models \phi \\
\mathcal{M}, \mathcal{A}, i \models \mathsf{X}\,\phi & \text{iff} & \mathcal{M}, \mathcal{A}, i + 1 \models \phi \\
\mathcal{M}, \mathcal{A}, i \models \phi \;\mathsf{U}\; \psi & \text{iff} & \exists j \geq i \,.\, (\mathcal{M}, \mathcal{A}, j \models \psi \;\wedge\; \forall i \leq k < j \,.\, \mathcal{M}, \mathcal{A}, k \models \phi) \\
\mathcal{M}, \mathcal{A}, i \models \phi \;\mathsf{R}\; \psi & \text{iff} & \forall j \geq i \,.\, (\mathcal{M}, \mathcal{A}, j \models \psi \;\vee\; \exists i \leq k < j \,.\, \mathcal{M}, \mathcal{A}, k \models \phi)
\end{array}
$$

$$
\begin{array}{rcl}
\mathcal{M}, \mathcal{A}, i \models \mathsf{H}\,\phi & \text{iff} & \forall 0 \leq j \leq i \,.\, \mathcal{M}, \mathcal{A}, j \models \phi \\
\mathcal{M}, \mathcal{A}, i \models \mathsf{O}\,\phi & \text{iff} & \exists 0 \leq j \leq i \,.\, \mathcal{M}, \mathcal{A}, j \models \phi \\
\mathcal{M}, \mathcal{A}, i \models \mathsf{Y}\,\phi & \text{iff} & i > 0 \wedge \mathcal{M}, \mathcal{A}, i - 1 \models \phi \\
\mathcal{M}, \mathcal{A}, i \models \phi \;\mathsf{S}\; \psi & \text{iff} & \exists 0 \leq j \leq i \,.\, (\mathcal{M}, \mathcal{A}, j \models \psi \;\wedge\; \forall j < k \leq i \,.\, \mathcal{M}, \mathcal{A}, k \models \phi) \\
\mathcal{M}, \mathcal{A}, i \models \phi \;\mathsf{T}\; \psi & \text{iff} & \forall 0 \leq j \leq i \,.\, (\mathcal{M}, \mathcal{A}, j \models \psi \;\vee\; \exists j < k \leq i \,.\, \mathcal{M}, \mathcal{A}, k \models \phi)
\end{array}
$$

SEMANTICS

$$
\begin{array}{rcl}
\mathcal{M}, \mathcal{A}, i \models \Phi \subseteq \Psi & \text{iff} & \mathcal{M}, \mathcal{A}, i \models \forall x_1, \ldots, x_{\mathrm{ar}(\Phi)}. \\
 & & \qquad \Phi(x_1, \ldots, x_{\mathrm{ar}(\Phi)}) \longrightarrow \Psi(x_1, \ldots, x_{\mathrm{ar}(\Phi)}) \\
\mathcal{M}, \mathcal{A}, i \models P(x_1, \ldots, x_n) & \text{iff} & (\mathcal{A}(x_1), \ldots, \mathcal{A}(x_n)) \in \pi(i)(P) \\
\mathcal{M}, \mathcal{A}, i \models \Phi'(t_1, \ldots, t_n) & \text{iff} & \mathcal{M}, \mathcal{A}, i+1 \models \Phi(t_1, \ldots, t_n) \\
 & \ldots &
\end{array}
$$