Lógica Primeira Ordem

- Proposições passam a predicados;
- Permitem quantificadores e que as variáveis variem perante o domínio;
- Conjuntos e Relações;
- M = (D,I)
  1. M -> Modelo
  2. D -> Domínio (não vazio) de todas as interpretações
  3. I -> Interpretação de constantes, funções e predicados

Alloy	Math
x_1 -> ... -> x_n **in** P	$P(x_1, \dots, x_n)$
x_1 -> ... -> x_n **not in** P	$\neg P(x_1, \dots, x_n)$
$x = y$	$x = y$
x != y	$\neg(x = y)$
not ϕ	$\neg\phi$
ϕ **and** ψ	$\phi \wedge \psi$
ϕ **or** ψ	$\phi \vee \psi$
ϕ **implies** ψ	$\phi \rightarrow \psi$
all $x : P \mid \phi$	$\forall x \cdot P(x) \rightarrow \phi$
some $x : P \mid \phi$	$\exists x \cdot P(x) \wedge \phi$

univ -> Topo
none -> Set vazio
iden -> Id

Alloy	Math
Φ **in** Ψ	$\Phi \subseteq \Psi$
Φ = Ψ	$\Phi = \Psi$
lone Φ	$\lvert\Phi\rvert \leq 1$
some Φ	$\lvert\Phi\rvert \geq 1$
no Φ	$\lvert\Phi\rvert = 0$
one Φ	$\lvert\Phi\rvert = 1$

Alloy	Math
$\Phi + \Psi$	$\Phi \cup \Psi$
$\Phi \& \Psi$	$\Phi \cap \Psi$
$\Phi - \Psi$	$\Phi \setminus \Psi$
$\Phi \text{ -> } \Psi$	$\Phi \times \Psi$
$\Phi . \Psi$	$\Phi . \Psi$
$A \text{ <: } \Phi$	$A \triangleleft \Psi$
$\Phi \text{ :> } A$	$\Phi \triangleright A$
$\sim \Phi$	Φ°
$\hat{} \Phi$	Φ^{+}
$* \Phi$	$\Phi^{\star}$
$\{ x : A \mid \phi \}$	$\{x \mid x \in A \wedge \phi\}$

abstract sig A{} -> Não existe um A que não seja uma extensão.
Extensões de assinaturas são sempre disjuntos.

Erros -> Slide nº3

Fun -> Servem para visualizar propriedades específicas.

LTL (Linear Temporal Logic)

- Fórmulas interpretadas por passos
- Tanto operadores de futuro como de passado
- Se uma fórmula LTL é satisfazível, então é satisfeita por pelo menos um “lasso trace” -> representação de traços infinitos usando traços finitos.

Electrum	Math	Meaning
always ϕ	G ϕ $\quad \Box \phi$	ϕ is always true from now on
eventually ϕ	F ϕ $\quad \Diamond \phi$	ϕ will eventually be true
after ϕ	X ϕ $\quad \bigcirc \phi$	ϕ will be true in the next state
ϕ **until** ψ	ϕ U ψ	ψ will eventually be true and ϕ is true until then
ϕ **releases** ψ	ϕ R ψ	ψ can only be false after ϕ is true

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

Bounded Model Checking (LTL)

- Traços infinitos;
- Traços infinitos representados por “lasso trace”;
- Para verificar uma fórmula, tenta-se encontrar um contra-exemplo;
- Relações mutáveis passam a K relações estáticas;
    - Fórmulas G (Always), F(eventually), são “desenroladas”;
    - Fórmulas X (After) verifica-se o estado atual e o imediatamente a seguir;

Propriedades Safety

- Algo de mau nunca irá acontecer;
- Basta verificar os prefixos (cuja continuação violam a propriedade) para verificar safety;
- Serve para limitar os comportamentos indesejados nos modelos;

```
assert safety{
        always ....
}
```

Propriedades Liveness

- Algo de bom irá eventualmente acontecer;
- Muito mais difícil de verificar comparativamente à safety, pois é necessário verificar todo o traço.

```
assert liveness{
        fairness implies eventually ....
}
```

Propriedade Fairness

- Necessária para verificar propriedades de liveness;
- Exclui traços onde um evento fica continuamente disponível mas nunca ocorre.
    - Strong Fairness -> Infinitely Often
    - Weak Fairness -> Permanently

Strong fairness

```
(always eventually enabled) implies (always eventually happens)
```

Weak fairness

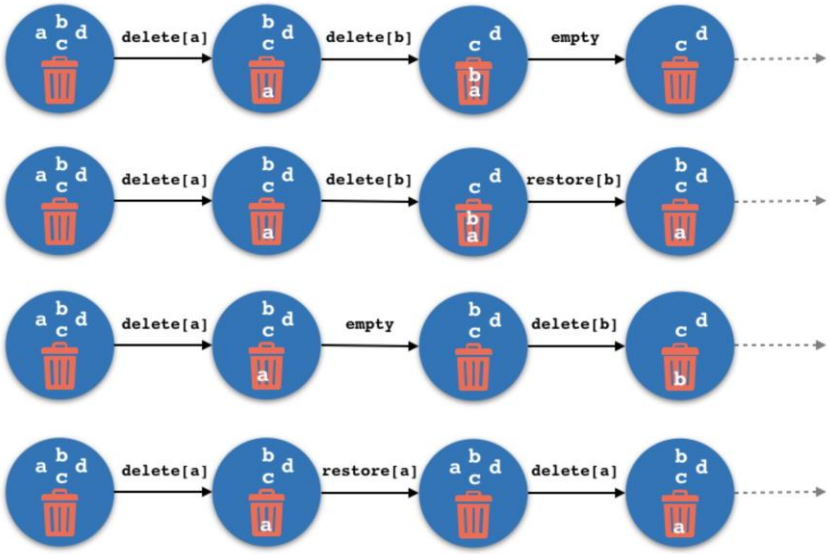
```
(eventually always enabled) implies (always eventually happens)
always ((always enabled) implies (eventually happens))
```

util/ordering[_]

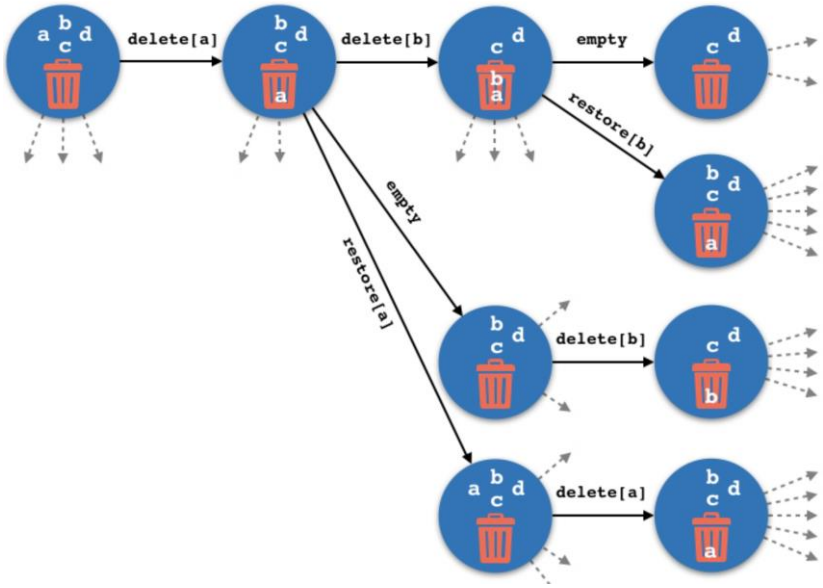
- next
- first
- last
- prev (~next)
- nextes
- prevs

Models of Time

- Linear Model of Time (LTL) (Ellectrum)
  - Tempo (lógico) é linearizado;
  - Sistema é representado por um conjunto infinito de traços;



- Branching Model of Time (CTL – Computational Tree Logic)
  - Em cada estado da árvore são mostradas todas as possibilidades a partir dos nodos;
  - Sistema é representado por um conjunto infinito de árvores de computação.



Operator	Meaning
G ϕ $\Box \phi$	ϕ is always true from now on
F ϕ $\Diamond \phi$	ϕ will eventually be true
X ϕ $\bigcirc \phi$	ϕ will be true in the next state
ϕ U ψ	ψ will eventually be true and ϕ is true until then
ϕ R ψ	ψ can only be false after ϕ is true

Operator	Meaning
A ϕ	ϕ is valid in all paths
E ϕ	ϕ is valid in some path

Slide 11 -> CTL Model Checking
Slide 12 -> LTL Model Checking