

Interpretadores Abstractos

Ernesto Rodriguez

Universidad del Itsmo

erodriguez@unis.edu.gt

¿Como se computa?

- Podemos definir funciones mediante *Tipos Abstractos*
- Sin embargo, necesitamos computar dichas funciones.
- A todo esto ¿Que es computar?

¿Como se computa?

- Podemos definir funciones mediante *Tipos Abstractos*
- Sin embargo, necesitamos computar dichas funciones.
- A todo esto ¿Que es computar?
 - Aplicar un conjunto de reglas para reducir un ADT.
 - La computación termina cuando ya no se puede aplicar ninguna regla

¿Como se computa?

- Dato $\mathcal{A} := \langle \mathcal{S}^0, \mathcal{D} \rangle$, llamamos al cuarteto $\langle f :: \mathbb{A} \rightarrow \mathbb{R}; \mathcal{R} \rangle$ un **procedimiento abstracto** ssi \mathcal{R} es un conjunto de reglas para f .
 - \mathbb{A} es el conjunto de parametros
 - \mathbb{R} es el conjunto de resultados
- La **computación** de un procedimiento abstracto p es la secuencia de **constructores** $c_1 \rightsquigarrow c_2 \rightsquigarrow \dots$ segun las reglas de p
- Una **computación abstracta** es una **computacion** que ejecutamos en nuestra mente sin limites de memoria o tiempo.
- Un **interpretador abstracto** es la maquina imaginaria que ejecuta dicha computación

Ejemplo: Las funciones ρ y concat

$$\left\langle \rho :: \mathcal{L}(\mathbb{N}) \rightarrow \mathcal{L}(\mathbb{N}); \quad \left\{ \begin{array}{l} \rho(\text{cons}(n, l)) \rightsquigarrow \text{concat}(\rho(l), \text{cons}(n, \text{nil})) \\ \rho(\text{nil}) \rightsquigarrow \text{nil} \end{array} \right\} \right\rangle$$

$$\left\langle \text{concat} :: \mathcal{L}(\mathbb{N}) \times \mathcal{L}(\mathbb{N}) \rightarrow \mathcal{L}(\mathbb{N}); \quad \left\{ \begin{array}{l} \text{concat}(\text{cons}(n, l), r) \rightsquigarrow \text{cons}(n, \text{concat}(l, r)) \\ \text{concat}(\text{nil}, r) \rightsquigarrow r \end{array} \right\} \right\rangle$$

Características de los Procedimientos Abstractos

- Se dice que un proceso abstracto **termina** si se produce un termino que ninguna regla en \mathcal{R} puede reducir
- Este modelo de computación esta inspirado en el calculo- λ de Church.
- Los “pasos”