# Cuantificadores

Matemática estructural y lógica

ISIS-1104

# Operadores binarios

Un operador binario sobre elementos de tipo $T$ es una función $T \times T \to T$

$$+ : Int \times Int \to Int$$

. Decimos que un operador $\oplus : T \times T \to T$ es:

- Asociativo si $(a \oplus b) \oplus c = a \oplus (b \oplus c)$
- Conmutativo si $(a \oplus b) = (b \oplus a)$
- Con $n$ como identidad si $(a \oplus n) = (n \oplus a) = a$

## Algunos ejemplos

- $+$ en *Int*, es asociativo, conmutativo y con $1$ como identidad.
- $/$ en *Real*, es no asociativo, no conmutativo y sin identidad.
- $\vee$ en *Bool*, es asociativo, conmutativo y con *False* como identidad.
- $\equiv$ en *Bool*, es asociativo, conmutativo y con *True* como identidad.

# Ahora ustedes!

- $-$ en *Int*?
- $*$ en *Int*?
- $\wedge$ en *Bool*?
- $\Rightarrow$ en *Bool*?

# Cuantificadores

Si $\oplus : T \to T \times T$ es conmutativo, asociativo y con identidad. Definimos su cuantificación como

$$(\oplus t : T \mid Q : E)$$

Lo cual se interpreta como

*sobre todos los t que cumplan $Q(t)$,*
*agregar el resultado de $E(t)$ usando $\oplus$*

$Q : T \to Bool$ se conoce como el *rango.*

$E : T \to T$ *se conoce como el cuerpo.*

# Cuantificadores vs. funciones de alto orden

En realidad un cuantificador es una combinación de funciones de alto orden

$$(\oplus t : T \mid Q : E$$

es lo mismo que

```
static T quantifier(⊕, Q, E, list) {
    List<T> filtered = filter(Q, list);
    List<T> mapped = map(E, filtered);
    T folded = fold(⊕, mapped);
    return folded;
}
```

Un cuantificador aplica sobre todos los elementos de tipo $T$ mientras que `quantifier` no.

# Un ejemplo

$$(+i : \mathit{Int} \mid i \geq 0 \land i < 6 : 2i)$$

Podemos pensarlo como

- Filtramos: Nos quedamos con los $i$ que cumplan $i \geq 0 \land i < 6$.

$$[0, 1, 2, 3, 4, 5]$$

- Mapeamos: Transformamos estos $i$ a $2i$

$$[0, 2, 4, 6, 8, 10]$$

- Reducimos: Operamos usando $+$

$$0 + 2 + 4 + 6 + 8 + 10 = 30$$

# Ahora ustedes

$$(* i : \mathit{Int} \mid i \geq 0 \land i < 10 \land \mathit{esPar}(i) : i/2)$$

# Algunos casos particulares

- Si no queremos filtrar hacemos el rango siempre verdadero:

$$(\oplus t : T \mid True : E \text{ o más corto } (\oplus t : T \mid : E)$$

- Las cuantificaciones pueden depender de otros valores:

$$(* i : Int \mid i > 0 \wedge i < n : i)$$

- Podemos operar infinitos elementos

$$(+ i : Int \mid i > 0 : {}^{1}/{2^{i}}) = \frac{1}{2^1} + \frac{1}{2^2} + \frac{1}{2^3} + ... = 2$$

- Pero no siempre tenemos que obtener un valor finito

$$(+ i : Int \mid i > 0 : i) \text{ diverge}$$

# ¿Cómo evaluar cuantificadores? Axiomas

- Rango vacío:

$$(\oplus t \mid False : E) = False$$

- Regla de un punto:

$$(\oplus t \mid t = s : E) = E(s)$$

- Distributividad*:

$$(\oplus t \mid Q : E_1) \;\oplus\; (\oplus t \mid Q(t) : E_2) = (\oplus t \mid Q : E_1 \oplus E_2)$$

# ¿Cómo evaluar cuantificadores? Axiomas

- Partir rango disyunto*: Si $Q \wedge S \equiv \mathit{False}$ para todos los $t$

$$(\oplus t \mid Q \vee S : E) = (\oplus t \mid Q : E) \ \oplus \ (\oplus t \mid S : E)$$

- Partir rango idempotente*: Si $t \oplus t = t$ para todos los $t$

$$(\oplus t \mid Q \vee S : E) = (\oplus t \mid Q : E) \ \oplus \ (\oplus t \mid S : E)$$

- Partir rango general*:

$$(\oplus t \mid Q \vee S : E) \oplus (\oplus t \mid Q \wedge S : E) = (\oplus t \mid Q : E) \oplus (\oplus t \mid S : E)$$

# ¿Cómo evaluar cuantificadores? Axiomas

- Intercambio de variables*: Si $Q$ no depende de $t$ y $S$ no depende de $v$

$$(\oplus t \mid Q(\oplus v \mid S : E) = (\oplus v \mid S(\oplus t \mid Q : E)$$

- Anidamiento*: Si $Q$ no depende de $v$

$$(\oplus t, v \mid Q \wedge S : E) = (\oplus t \mid Q : (\oplus v \mid S : E))$$

- Renombramiento de variables: Si $Q$ y $E$ no dependen de $v$

$$(\oplus t \mid Q : E) = (\oplus v \mid Q : E)$$