Razonamiento Ecuacional sobre Estructuras de Datos.

Pilar Selene Linares Arévalo

Facultad de Ciencias, UNAM. México.

L Congreso Nacional de la Sociedad Matemática Mexicana
Octubre 2017.

Este trabajo se realiza con el apoyo del proyecto UNAM PAPIME PE102117.

Consideremos la proposición

$$p \wedge q \equiv q \wedge p$$

¿Qué significa que esta proposición sea válida? ¿Es posible dar un contraejemplo?

Ahora consideremos

$$x + y = y + x$$

¿Es posible dar un contraejemplo?

Necesitamos un mecanismo diferente para demostrar que una proposición como la anterior es válida.

Únicamente a partir de las ecuaciones:

$$\begin{aligned} x + y &= y + x & (1) \\ x + (y + z) &= (x + y) + z & (2) \end{aligned}$$

podemos concluir que $w_1 + (w_2 + w_3) = (w_3 + w_2) + w_1$
pero no podemos afirmar que $w_1 + w_1 = w_1 + w_2$.

- ¿Cuándo una igualdad es consecuencia lógica de un conjunto de ecuaciones?

- ¿Cómo deducimos nuevas igualdades?

$$\begin{aligned} w_1 + (w_2 + w_3) &= w_1 + (w_3 + w_2) \\ &= (w_3 + w_2) + w_1 \end{aligned}$$

La Lógica Ecuacional es una parte de la Lógica de Primer Orden que trabaja exclusivamente con expresiones en forma de ecuaciones.

$$\frac{}{t = t}\ \textsc{Refl} \qquad \frac{s = t}{t = s}\ \textsc{Sym} \qquad \frac{t = r \qquad r = s}{t = s}\ \textsc{Trans}$$

$$\frac{t = s}{t\sigma = s\sigma}\ \textsc{Inst} \qquad \frac{t_1 = s_1 \quad \ldots \quad t_n = s_n}{f(t_1, \ldots, t_n) = f(s_1, \ldots, s_n)}\ \textsc{Congr}$$

$$\frac{t = s \qquad E[x := s]}{E[x := t]}\ \textsc{Rewrite}$$

$$\frac{t = s}{E[x := t] = E[x := s]}\ \textsc{Leibniz}$$

Sea

$$\Gamma = \{ \begin{array}{rcl} cuadrado(x) & = & mult(x,x) \\ cubo(x) & = & mult(x, cuadrado(x)) \\ cuarta(x) & = & mult(cuadrado(x), cuadrado(x)) \end{array} \}$$

¿ $\Gamma \vdash cuarta(3) = mult(mult(3,3), mult(3,3))$?

$$\begin{array}{rcll} cuarta(x) & = & mult(cuadrado(x), cuadrado(x)) & \text{Dada} \\ cuarta(3) & = & mult(cuadrado(3), cuadrado(3)) & \text{inst [x:=3]} \\ cuadrado(x) & = & mult(x,x) & \text{Dada} \\ cuadrado(3) & = & mult(3,3) & \text{inst [x:=3]} \\ cuarta(3) & = & mult(mult(3,3), mult(3,3)) & \text{Rewrite} \end{array}$$

$\therefore \Gamma \vdash cuarta(3) = mult(mult(3,3), mult(3,3))$

Tipo de Dato Abstracto: Cola

```
vacia : Q t
insertar : t -> Q t -> Q t
eliminar : Q t - > Q t
frente : Q t - > t
```

frente(insertar α vacia) = α

frente(insertar β (insertar α q)) = frente (insertar α q)

eliminar(insertar α vacia) = vacia

eliminar(insertar β (insertar α q)) = insertar β (eliminar (insertar α q))

$\therefore$ frente(elim(ins z (elim (ins y (ins x vacia))))) = z

$$
\begin{aligned}
frente(elim\ (ins\ z\ (elim\ (ins\ y\ (ins\ x\ vacia))))) &= z && \triangleright \\
frente(elim\ (ins\ z\ (ins\ y\ (elim\ (ins\ x\ vacia))))) &= z && \triangleright \\
frente(elim\ (ins\ z\ (ins\ y\ (vacia)))) &= z && \triangleright \\
frente(ins\ z\ (elim\ (ins\ y\ vacia))) &= z && \triangleright \\
frente(ins\ z\ (vacia)) &= z && \triangleright
\end{aligned}
$$

Consideremos el siguiente programa en HASKELL:

```
map f [] = []
map f (x:xs) = f x : map f xs

(f.g) x = f (g x)
```

Queremos mostrar que el programa cumple la siguiente propiedad:

$$map\ f'\ (map\ g'\ l) = map\ (f'.g')\ l$$

Inducción sobre la lista l. Caso Base:

$$\begin{array}{rcll} map\ f'\ (map\ g'\ [\,]) & = & map\ (f'.g')\ [\,] & \triangleright \\ map\ f'\ ([\,]) & = & map\ (f'.g')\ [\,] & \triangleright \\ [\,] & = & map\ (f'.g')\ [\,] & \triangleright \\ [\,] & = & [\,] & \square \end{array}$$

Consideremos el siguiente programa en HASKELL:

```
map f [] = []
map f (x:xs) = f x : map f xs

(f.g) x = f (g x)
```

Queremos mostrar que el programa cumple la siguiente propiedad:

$$map\ f'\ (map\ g'\ l) = map\ (f'.g')\ l$$

Hipótesis de inducción:

$$map\ f'\ (map\ g'\ xs) \quad = \quad map\ (f'.g')\ xs$$

Consideremos el siguiente programa en HASKELL:

```
map f [] = []
map f (x:xs) = f x : map f xs

(f.g) x = f (g x)
```

Queremos mostrar que el programa cumple la siguiente propiedad:

$$map\ f'\ (map\ g'\ l) = map\ (f'.g')\ l$$

Paso inductivo:

$$\begin{array}{rcll}
map\ f'\ (map\ g'\ x{:}xs) & = & map\ (f'.g')\ x{:}xs & \triangleright \\
map\ f\ (g'\ x : (map\ g'\ xs)) & = & map\ (f'.g')\ x{:}xs & \triangleright \\
f'\ g'\ x : map\ f'\ (map\ g'\ xs) & = & map\ (f'.g')\ x{:}xs & \triangleright \\
f'\ g'\ x : map\ f'.g'\ xs & = & map\ (f'.g')\ x{:}xs & \triangleright \\
f'\ g'\ x : map\ f'.g'\ xs & = & (f'.g')\ x : map\ (f'.g')\ xs & \triangleright \\
f'\ g'\ x : map\ f'.g'\ xs & = & f'\ g'\ x : map\ (f'.g')\ xs & \square
\end{array}$$

Prueba usual

$$\begin{aligned} \mathit{cuarta}(x) &= \mathit{mult}(\mathit{cuadrado}(3), \mathit{cuadrado}(3)) \\ &= \mathit{mult}(\mathit{mult}(3,3), \mathit{mult}(3,3)) \end{aligned}$$

Prueba con Razonamiento Ecuacional

$$\begin{array}{rcll} \mathit{cuarta}(x) & = & \mathit{mult}(\mathit{cuadrado}(x), \mathit{cuadrado}(x)) & \text{Dada} \\ \mathit{cuarta}(3) & = & \mathit{mult}(\mathit{cuadrado}(3), \mathit{cuadrado}(3)) & \text{inst [x:=3]} \\ \mathit{cuadrado}(x) & = & \mathit{mult}(x,x) & \text{Dada} \\ \mathit{cuadrado}(3) & = & \mathit{mult}(3,3) & \text{inst [x:=3]} \\ \mathit{cuarta}(3) & = & \mathit{mult}(\mathit{mult}(3,3), \mathit{mult}(3,3)) & \text{Rewrite} \end{array}$$

¿Por qué usar razonamiento ecuacional si las demostraciones parecen ser más complejas?

Consideremos la siguiente definición de la estructura algebraica de anillo:

Anillo

Un anillo es un conjunto R junto con dos operaciones $(R, +, \cdot)$, un elemento distinguido 0 y una operación unaria $^{-1}$ que satisfacen los siguientes axiomas:

$$\begin{array}{rclcrcll}
x + y & = & y + z & A1 & x^{-1} + x & = & 0 & A4 \\
(x + y) + z & = & x + (y + z) & A2 & (x \cdot y) \cdot z & = & x \cdot (y \cdot z) & M1 \\
0 + x & = & x & A3 & x \cdot (y + z) & = & x \cdot y + x \cdot z & D1
\end{array}$$

¡Gracias!