

## Introducción

En matemática se utiliza el símbolo  $=$  con dos significados distintos: definir un objeto, y averiguar si dos objetos son iguales o no. En un lenguaje de programación esto es inaceptable, por lo que hemos dicho anteriormente: el computador es incapaz de deducir de cuál significado se trata. Distintos lenguajes de programación utilizan distintos símbolos para uno y otro significado. En ISetL, se define con  $:=$  y se averigua con  $=$ .

## La definición

Una definición asocia un nombre con un objeto matemático. El siguiente es una definición muy simple para "x":

$$x := 2$$

A partir de ahora, y hasta que no se "redefina", el nombre "x" será otra forma para referir al objeto 2.

Consideremos ahora las siguientes dos definiciones

$$x := 2$$
$$y := x + 1$$

La variable "x" nombra al 2, mientras que la variable "y" nombra al resultado de evaluar la expresión "x+1". Es decir, el nombre "y" será otra forma de referirnos al objeto 3.

Ahora, consideremos la siguiente secuencia de definiciones

$$x := 2$$
$$y := x + 1$$
$$x := 1$$

Al terminar de ejecutar las definiciones (una sesión), "x" nombrará al valor 1, mientras que "y" nombrará al valor 3. La variable "x" ha sido redefinida, pero este cambio no afecta la asociación ya establecida entre la variable "y" y el valor 3.

Una definición tiene la forma "nombre  $:=$  objeto". En las expresiones de arriba, los objetos son constantes enteras y expresiones aritméticas. En algunas de las definiciones de la Actividad 1, hemos usado como objetos variables booleanas y proposiciones. Por ejemplo, en  $p := x \text{ in } A$ , el nombre  $p$  se asocia al valor de la proposición  $x \text{ in } A$  que es true o false ( $x$  y  $A$  están definidas previamente). Al ejecutarse la definición, se asocia al nombre la forma canónica del objeto (o de la expresión).

## Proposiciones

Las proposiciones son expresiones que tienen por forma canónica valores del conjunto **Bool**, definido como

$\text{Bool} := \{\text{true}, \text{false}\};$

Para las siguientes definiciones en ISetL

$x := 2;$

$A := \{0, 2, 4, 6\};$

$p := x \text{ in } A;$

comprobamos que el valor de  $p$  es true, (verdadero).

Las variables asignadas a proposiciones, son las variables proposicionales.

Por ejemplo, **p** arriba es una variable proposicional y **x in A** es una proposición. Las constantes **true** y **false** y las variables proposicionales también se consideran proposiciones.

Las proposiciones no canónicas se reducen a otras aplicando las operaciones comúnmente conocidas. Algunas operaciones están predefinidas en ISetL y se denotan como sigue: ( $p$  y  $q$  son variables proposicionales)

Lenguaje natural	ISetL
-----	
no $p$ ,	not $p$ ,
$p$ y $q$ ,	$p$ and $q$ ,
$p$ o $q$ ,	$p$ or $q$ ,
$p$ implica $q$	$p \text{ impl } q$
$p$ es equivalente a $q$	$p = q$
menor, mayor, etc	$<, >, \leq, \geq, \neq$

$p$  implica  $q$  es también si  $p$  entonces  $q$  y se suele escribir en matemáticas como  $p \rightarrow q$

## Ejemplos

$p := (32 \bmod 15) = 2;$

$q := 32 < 15;$

not  $p$  and ( $p \text{ impl } q$ );