Lógica de Predicados. Introducción.

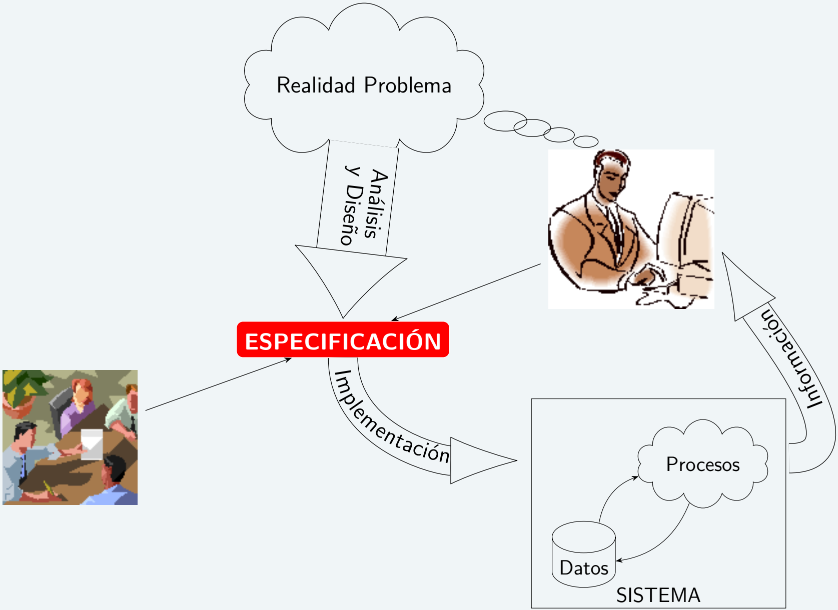
Lógica

Motivación

Sistema vs Realidad

- Un sistema informático no es otra cosa que un **modelo de una parte de la realidad**, típicamente de un servicio.
  - el servicio que debe proveer la bedelía de la facultad o un banco o un supermercado, etc.
- Cómo se construye típicamente este modelo?

Sistema Informático



Especificación

- Documento que refleja el acuerdo entre el usuario y el equipo de desarrollo sobre lo que debe hacer o no un sistema.
- Documento que refleja el acuerdo entre los integrantes del equipo de desarrollo sobre qué representa cada dato y qué debe hacer cada módulo, función, etc.
- Es un modelo donde los objetos que se especificaron se comportan de forma similar a los objetos reales.
- ***Si no se dispone de un mecanismo adecuado para formalizar hasta cierto punto la realidad, no es posible construir un sistema informático que la modele***.

Lenguajes de Especificación

- La especificación debe proveer lo necesario para realizar las tareas básicas que se hacen con ella:
  - Describir el problema sin ambigüedad.
  - Construir una solución adecuada del problema y con un trabajo razonable.
  - Verificar la solución que se construyó con respecto a la descripción.
- Dependiendo de la claridad de la definición de la sintaxis y semántica del lenguaje de especificación, ésta será más o menos formal.

Los objetos se comportan como los reales

Lenguajes de Especificación

- El lenguaje que se usa para construir las especificaciones debe cumplir algunas características, entre ellas:
    - Permitir la referencia a los elementos del problema.
    - Permitir la identificación de diferentes clases de elementos.
    - Poder ser utilizado en diferentes contextos o al menos diferentes problemas.

Prop como Lenguaje de Especificación: Ordenar un Array

- Dado un array de enteros, devolver otro ordenado con los mismos valores.
- Correspondencia.
  - A es un array: P.
  - El programa (función) Ordenar funciona bien: R.
  - B es la salida de Ordenar: Q.
  - A y B son permutaciones de un mismo array: S.
- Especificación.
  - $P \wedge R \wedge Q \rightarrow S$
- **Quién garantiza que A y B están relacionados de alguna forma?**

Prop como Lenguaje de Especificación: Conclusiones

- Prop no es un buen lenguaje de especificación ya que sólo permite hacer referencia a las nociones de **verdadero** y **falso**.
- Esto puede tener su contexto de aplicación.
  - Ej. Electrónica Digital
- Para especificar en informática, es necesario hacer referencia a elementos de la realidad.
  - Ej: edades, personas, asignaturas, bolsas de arroz, etc.

Buscando otro Lenguaje

- Pensemos en utilizar el metalenguaje usado en el curso.
- Qué significa que un array está ordenado?

$$
\begin{aligned}
Ordenado(b) \equiv (\bar{\forall} i. \\
i \in \mathcal{N} \text{ y } 1 \leq i \leq (len(b) - 1) \\
\Rightarrow b[i] \leq [i + 1])
\end{aligned}
$$

- Qué significa que dos arrays tienen los mismos elementos?

$$
\begin{aligned}
TME(a, b) \equiv Incluido(a, b) \text{ y } Incluido(b, a) \\
Incluido(a, b) \equiv \bar{\forall} i. i \in \mathcal{N} \text{y} 1 \leq i \leq len(a) \\
\Rightarrow \bar{\exists} j : j \in \mathcal{N} y 1 \leq j \leq len(b) : a[i] = b[j]
\end{aligned}
$$

Buscando otro Lenguaje

- Dado que la función pedida tiene que cumplir con las dos condiciones, f va a resolver lo pedido si está en el siguiente conjunto:

$$\{f / f : ArrayInt \to ArrayInt \wedge \\ \forall a : a \in ArrayInt : (Ordenado(f(a)) \\ \wedge TME(a, f(a))) \\ \}$$

- La especificación representa el conjunto de soluciones al problema.

Especifica o no?

- Eliminación de la ambigüedad?
  - Si, porque a pesar que no es formal dado que sólo son abreviaturas del idioma español, hay un acuerdo con respecto al significado.
- Construir una solución adecuada con un trabajo razonable?
  - Si, si somos capaces de construir un elemento del conjunto que se especificó, en algún lenguaje dado, por ejemplo, Módula.
- Referencia a los elementos de la realidad?
  - Si.

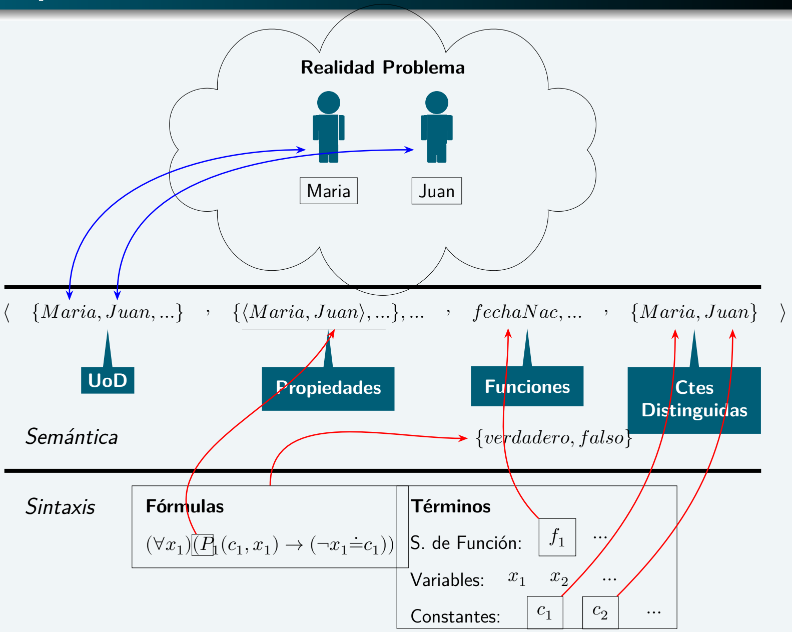
Especifica o No?

- Puede ser utilizado en diferentes contextos o al menos diferentes problemas?
  - Si, lo hemos estado utilizando en todo el curso para diferentes cosas.
- Permite verificar la solución que se construyó con respecto a la descripción?
  - Parece que si... Para hacerlo bien es necesario formalizar mejor el propio lenguaje e incluso su manipulación.
- La idea es construir un sistema similar al de Prop pero para un lenguaje como este.

Análisis e Interpretación del Lenguaje

- En la condición que define al conjunto hay dos tipos de elementos:
  - Unos que referencia a array's o enteros (ej. **a**, **b**, **0**, **len(a)**).
  - Otros que referencian a propiedades o relaciones que deben cumplir esos elementos (ej. **i** $<$ **len(b)-1**, o **Ordenado(f(a))**).
- Los primeros referencia a elementos de un ***Universo de Discurso (UoD)*** dado
- Los segundos son una forma de expresar ***hechos que pueden ser verdaderos o falsos*** dependiendo de ese universo y la interpretación que se les dé a los símbolos.

Lo que Vendrá



Lo que Vendrá

- Sintaxis de los Lenguajes de Primer Orden.
  - Se definirán los términos y las fórmulas como conjuntos inductivos.
- Semántica de los Lenguajes de Primer Orden.
  - Se definirán formalmente las funciones que hacen la correspondencia de la sintaxis con la semántica y se estudiarán propiedades de esas correspondencias.
- Deducción Natural en Primer Orden.
  - Se definirán reglas que nos permitirán construir derivaciones sin involucrar la semántica.
- Completitud y sus aplicaciones en Primer Orden.
  - Se estudiarán las propiedades de completitud y corrección del sistema definido anteriormente

Contenidos

- Calculo de Predicados

Cálculo Proposicional

Formalización en PROP:

$$\overbrace{\textbf{Todo natural es entero}}^{p}$$

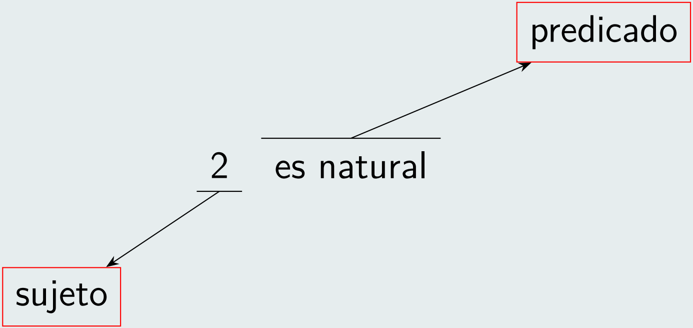
$$\textbf{Si } \overbrace{\textbf{2 es un natural}}^{q}\textbf{, entonces } \overbrace{\textbf{2 es un entero}}^{r}.$$

- Pero sin embargo:

$$p \not\models q \rightarrow r$$

- **Necesitamos un formalismo más expresivo**

- La validez de ciertos razonamientos depende de la relación entre las proposiciones
- Solución: Hacer análisis fino de la estructura de las proposiciones



Predicados

« 2 es natural »

En lógica proposicional: p
(una prop. atómica)

En predicados: Natural(2)
Sujeto: 2
Propiedad: Ser Natural

Lenguaje de La Lógica de Predicados

- Símbolos para denotar objetos
- Símbolos para denotar propiedades y relaciones
- Conectivos
- Cuantificadores

Símbolos para Denotar Objetos

- Símbolos de constante: permiten referirse a objetos determinados
  - Mafalda, 2, π
- Símbolos de variable: permiten referirse a objetos genéricos
  - x, n, α
- Símbolos de función: permiten referirse a operaciones (unarias, binarias, etc.)
  - $m + 1, 2!, (1 + 1)!$

Símbolos de Predicado

- Permiten representar propiedades y relaciones entre objetos (símbolos unarios, binarios, etc.)
  - Par es un símbolo de propiedad (unario)
  - $\geq$ es un símbolo de relación binario
- Los símbolos de predicado se aplican a objetos para representar afirmaciones simples:
  - Par(2)
  - $x \geq 1$

Conectivos

- Permiten combinar afirmaciones.
- Igual que en lógica proposicional:
  - $\bot, \wedge, \vee, \neg \rightarrow, \leftrightarrow$
- Ejemplos:
  - $Par(2) \wedge x \geq 1$
  - $x \geq 1 \rightarrow \neg\bot$

Cuantificadores

- Cuantifican los objetos genéricos (variables)
  - Cuantificador Universal: $\forall$
  - Cuantificador Existencial: $\exists$
- Ejemplos
  - $(\forall n)((Par(n) \wedge 1 \geq n) \rightarrow n = 0)$
  - $(\forall x)(\exists y)x \geq y$

Ejemplos

- El factorial de todo número es par
  - $(\forall x)Par(x!)$
- La suma de dos pares es par
  - $(\forall x)(\forall y)(Par(x) \wedge Par(y) \rightarrow Par(x+y))$
- Todo número natural es par o impar
  - $(\forall n)(Par(n) \vee Impar(n))$
- Ningún número es a la vez par e impar
  - $\neg(\exists x)(Par(x) \wedge Impar(x))$
- Todo número natural par tiene raíz cuadrada
  - $(\forall n)(Par(n) \rightarrow (\exists m)(m * m = n))$

Universo de discurso

- En matemática usamos algunas convenciones informales para indicar dominios:
  - naturales: n, m, k
  - reales: x, y, z
  - fórmulas lógicas: α, β, γ
  - Conjuntos de fórmulas: Γ, Δ
- En Lógica de predicados los objetos pertenecen todos a un mismo universo.
  - No hay forma de diferenciar sintácticamente los distintos dominios

Universo de discurso

- Cuando es necesario particionar el universo de discurso en clases de objetos, utilizamos símbolos de propiedad para referenciar los objetos de la subclase:
  - Todo natural es par o impar:$(\forall x)(N(x) \rightarrow Par(x) \vee Impar(x))$
- Si la naturaleza de los objetos de quienes hablamos está sobreentendida (ej. hablamos siempre de fórmulas, naturales, reales, etc.) podemos obviar el símbolo de propiedad respectivo

Símbolos

- ¿Qué determina los símbolos del alfabeto que necesitamos en nuestro lenguaje?
  - Ningún número es par e impar a la vez:
    $\neg(\exists x)(Par(x) \wedge Impar(x))$
    $\neg(\exists x)(Par(x) \wedge \neg Par(x))$
- La Estructura: depende de la realidad que queremos describir.