Conceptos Básicos
Lógica

Bases de Datos para Ingeniería
-
Bases de Datos y Sistemas de Información

Instituto de Computación - Facultad de Ingeniería

Contenido

- Motivación
- Especificación
- Lenguaje de Especificación
- Lenguaje de la Lógica de predicados
- Valores, variables y tipos
- Valores nulos

Motivación

Motivación

- En el mundo real, las consultas a una Base de Datos se basan y se parecen mucho a la lógica de predicados.

- 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?

Especificación

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

Lenguajes de especificación

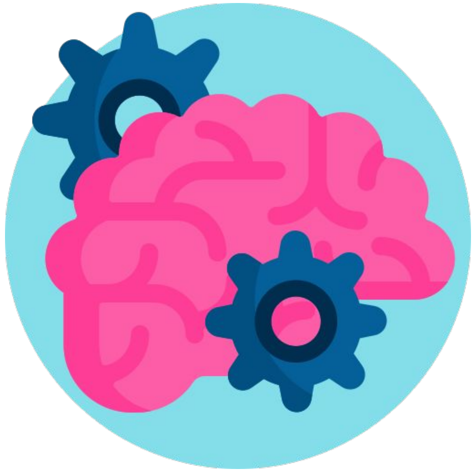
- La especificación debe proveer lo necesario para realizar las tareas básicas que se hacen con ella:
  - Describir el problema sin ambigüedad.
  - Construir una solución adecuada del problema y con un trabajo razonable (Los objetos se comportan como los reales).
  - Verificar la solución que se construyó 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.

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.

- Para especificar en informática, es necesario hacer referencia a elementos de la realidad.
  - Ejemplo: edades, personas, asignaturas, bolsas de arroz, etc.

Lenguaje de Lógica de la Predicados



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

Lenguaje de la Lógica de Predicados

- **Símbolos para denotar objetos**
  - **Símbolos de constante:** permiten referirse a objetos determinados. Representan valores que no varían a lo largo del tiempo.
    - Ejemplo: Mafalda, 2, p
  - **Símbolos de variable:** permiten referirse a objetos genéricos. Representan valores que varían a lo largo del tiempo.
    - Ejemplo: x, n, a
  - **Símbolos de función:** permiten referirse a operaciones (unarias, binarias, etc.)
    - Ejemplo: m+1, 2! , (1+1)!

Valores, variables y tipos

- Un **Valor** es una constante individual, con un significado bien definido
    - Ejemplo: 17

- Un **Valor** no se puede modificar, ya que no sería el mismo Valor.

- Una **Variable** mantiene un **Valor**, pero puede mantener múltiples Valores a lo largo del tiempo.

- Las **Variables** sí se pueden modificar; cambiando el Valor que mantienen.

- Las **Variables** tienen nombre, lo que nos permite hablar (o predicar) sobre ellas.

Valores, variables y tipos

- Todas las Variables tienen un **Tipo**

- El **Tipo** de una Variable es el conjunto de todos los Valores que la Variable puede mantener.
  - Ejemplos comunes de tipo: strings (cadena de caracteres), números enteros o fechas.

- Una base de datos se puede ver como una variable. En cada momento del tiempo tiene un determinado valor (de un tipo muy complejo, determinado por el esquema de la base de datos)

Lenguajes de la Lógica de Predicados

- **Símbolos de predicados:**
  - Permiten representar propiedades y relaciones entre objetos (símbolos unarios, binarios, etc.)
    - Ejemplo Par(2): “2 es un número par”, Par es un símbolo de propiedad (unario)
    - > es un símbolo de relación binario
  - Los símbolos de predicado se aplican a objetos para representar afirmaciones simples:
    - Par(2)
    - $x > 1$

Lenguaje de la Lógica de Predicados

- **Conectivos**: Permiten combinar afirmaciones:
  - Algunos conectivos: not ($\neg$), entonces ($\rightarrow$), sí y sólo sí ($\leftrightarrow$), or ($\vee$), and ($\wedge$)
    - Ejemplo: $Par(2) \wedge (x > 1)$
- **Cuantificadores**: Cuantifican los objetos genéricos (variables)
  - Cuantificador Universal (para todo): $\forall$
  - Cuantificador Existencial (existe): $\exists$
- Ejemplos:
  - $(\forall n)\ ((Par(n) \wedge (1 >= n)) \rightarrow n = 0)$
  - $(\forall x)\ (\exists y)\ x > y$

Tablas de verdad

Tablas de Verdad

AND

A	B	A ∧ B
V	V	V
V	F	F
F	V	F
F	F	F

OR

A	B	A ∨ B
V	V	V
V	F	V
F	V	V
F	F	F

NOT

A	¬A
V	F
F	V

Proposiciones y Predicados

Proposiciones y Predicados

- Las proposiciones tienen un valor de verdad, que puede ser **Verdadero** o **Falso**.

- Un ejemplo de proposición es "1 + 1 = 3", la cual obviamente, es falsa...

- Un predicado, a diferencia de una proposición, admite variables de las que no se conoce su valor de verdad.

- Un ejemplo de predicado es "X aprobará Matemáticas“. El valor de verdad de este predicado depende de la variable X, que varía en el conjunto de estudiantes.

- **Instanciar un predicado con valores:** Si se sustituye la variable X por el nombre “José” se obtiene una proposición.

Proposiciones y Predicados

- Hay proposiciones, y de la misma forma predicados, que siempre son verdaderas y se llaman **tautologías**.
    - Ejemplo: $1 = 1$ es una tautología

- Hay proposiciones, y de la misma forma predicados, que siempre son falsas y se llaman **contradicciones**.
    - Ejemplo: $1 = 2$ es una contradicción

Lenguaje de la Lógica de Predicados

- Noción de fórmula
    - La variable x es un término
    - La constante c es un término
    - Si t1,... tn son términos entonces P(t1,...,tn) es una fórmula
    - Si α y β son fórmulas, entonces
        - $(\alpha \rightarrow \beta)$ es una fórmula
        - $(\alpha \leftrightarrow \beta)$ es una fórmula
        - $(\alpha \wedge \beta)$ es una fórmula
        - $(\alpha \vee \beta)$ es una fórmula
    - $\neg \alpha$ es una fórmula
    - $((\forall x) \alpha)$ es una fórmula
    - $((\exists x) \alpha)$ es una fórmula

Lenguaje de la Lógica de Predicados

- **Atención:** No confundir las siguientes fórmulas, tener en cuenta los paréntesis
  - $(\forall x)(\alpha \rightarrow \beta)$ y $(\forall x)\ \alpha \rightarrow \beta$
  - $(\exists x)(\alpha \rightarrow \beta)$ y $(\exists x)\ \alpha \rightarrow \beta$
- Alcance de cuantificadores
  - El alcance del cuantificador $\forall x$ en la fórmula $((\forall x)\ \alpha)$ es la fórmula α
  - El alcance del cuantificador $\exists x$ en la fórmula $((\exists x)\ \alpha)$ es la fórmula α
- Ejemplos:
  - $(\forall x1)P1(x1) \rightarrow (\forall x2)P2(x1,x2)$
  - $(\forall x2)(\forall x1)(P1(x1) \rightarrow P2(x1,x2))$

Lenguaje de la Lógica de Predicados

- Atención: No confundir las siguientes fórmulas, tener en cuenta los paréntesis
  - $(\quad x)(\alpha \rightarrow \beta)$ y $(\quad x)\ \alpha \rightarrow \beta$
  - $(\exists x)(\alpha \rightarrow \beta)$ y $(\exists x)\ \alpha \rightarrow \beta$

- Alcance de cuantificadores
  - El alcance del cuantificador $\quad x$ en la fórmula $((\quad x)\ \alpha)$ es la fórmula α
  - El alcance del cuantificador $\exists x$ en la fórmula $((\exists x)\ \alpha)$ es la fórmula α

- Ejemplos:
  - $(\quad x_1)P_1(x_1) \rightarrow (\quad x_2)P_2(x_1,x_2)$
  - $(\quad x_2)(\quad x_1)(P_1(x_1) \rightarrow P_2(x_1,x_2))$

Lógica de Predicados: Ejemplos

- 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
    - $(\forall n)\ (\ Par(n) \rightarrow (\exists m)\ (m*m = n)\ )$

Valores nulos

- Se admite la existencia de un valor que no es verdadero ni falso, que representa la ausencia de información, porque es desconocida o simplemente porque no aplica. Este valor se llama NULL.

- El valor desconocido es un valor que aplica a todos los tipos de datos y es de naturaleza diferente al resto de los valores del tipo. Por ejemplo, NULL no es igual al string vacío o al número entero 0 (se codifica de una forma diferente).

Ejercicio

Ejercicio

- not (¬)
    - El sillón es cómodo
        - Cómodo(sillón)
    - El sillón es incómodo
        - ¬ (Cómodo(sillón))

Ejercicio

- entonces (➔)
    - Si Rebeca es campeona de natación entonces sabe nadar
        - Campeona_Natación(Rebeca) ➔ Nada(Rebeca)
- sí y solo sí (↔)
    - Rebeca es la tía de Silvio
        - Tía(Rebeca,Silvio) ↔ Sobrino(Silvio,Rebeca)

Ejercicio

- or (∨)
  - Rebeca es ingeniera o matemática
    - Matemática(Rebeca) ∨ Ingeniera(Rebeca)
- and (∧)
  - Silvio es pianista y pintor
    - Pianista(Silvio) ∧ Pintor(Silvio)

Ejercicio

- para todo ($\forall$)
  - Todas las liebres son rápidas
    - $(\forall x)$ (Liebre(x) → Rápida(x))
- existe ($\exists$)
  - Alguna liebre lenta seguro hay
    - $(\exists x)$ (Liebre(x) ∧ Lenta(x))

Ejercicio

- ¿Cómo especificaría lo siguiente?

Si Rebeca es matemática o ingeniera, tiene un título universitario