Lógica de Predicados

Prof. Sergio Salinas

Licenciatura en Ciencias de la Computación
Facultad de Ingeniería
Universidad Nacional de Cuyo

Lógica - 2025

Outline

Introducción

Limitaciones lógica proposicional

Limitaciones de la lógica proposicional:

- Incapacidad para expresar propiedades de objetos o relaciones entre ellos.
- Falta de operadores de cuantificación.
- Ausencia de variables para representar el dominio de un problema.
- Limitada capacidad de abstracción.

Historia

- La lógica de predicados, también conocida como lógica de primer orden, es una extensión de la lógica proposicional que introduce el uso de cuantificadores y predicados para hablar de propiedades de objetos y relaciones entre ellos de manera más precisa.

- La lógica de predicados tiene sus raíces en la lógica aristotélica, desarrollada por Aristóteles en el siglo IV a.C. Aristóteles introdujo el silogismo, un tipo de argumento lógico que se convirtió en la base de la lógica occidental durante casi dos milenios.

- En el siglo XIX, los matemáticos comenzaron a buscar fundamentos más sólidos para la matemática, lo que llevó al desarrollo de la lógica simbólica. George Boole y Augustus De Morgan desarrollaron la lógica algebraica que eventualmente se convertiría en la lógica proposicional. Sin embargo, aún estaba limitada en su capacidad para expresar relaciones generales.

Historia

- Gottlob Frege, un matemático, lógico y filósofo alemán, dio un salto cualitativo en 1879 con la publicación de su obra "Begriffsschrift" (Escritura Conceptual), que es considerada la primera formulación de la lógica de predicados. Frege introdujo un sistema formal que incluía cuantificadores y variables, permitiendo expresiones más complejas que las posibles en la lógica proposicional.
- A principios del siglo XX, Alfred North Whitehead y Bertrand Russell publicaron "Principia Mathematica" (1910-1913), un intento de derivar toda la matemática conocida usando la lógica como fundamento. Aunque el proyecto no logró su objetivo último debido a problemas como el teorema de incompletitud de Gödel, estableció la lógica de predicados como un lenguaje fundamental para la matemática.

Historia

- En la década de 1920, David Hilbert propuso su programa para proporcionar fundamentos sólidos a toda la matemática, parte del cual implicaba el desarrollo de sistemas lógicos completos y consistentes. Aunque el programa de Hilbert enfrentó desafíos insuperables debido a los teoremas de incompletitud de Gödel, contribuyó significativamente al estudio de la lógica formal.

- Kurt Gödel, en 1931, demostró sus famosos teoremas de incompletitud, que tuvieron profundas implicaciones para la lógica formal y la fundamentación de la matemática. Gödel utilizó la lógica de predicados en sus demostraciones, mostrando tanto su poder como sus límites.

Historia

Lógica de predicados

La lógica de predicados ha influido profundamente en varias áreas, desde la filosofía hasta la informática. Ha sido crucial para el desarrollo de la teoría de modelos, la informática teórica, la programación lógica, y la inteligencia artificial, particularmente en el diseño de lenguajes de programación y sistemas de bases de datos.

Introducción

Definición

La lógica de predicados, también conocida como lógica de primer orden, es una extensión de la lógica proposicional que permite expresiones más complejas mediante el uso de cuantificadores, predicados y objetos. Proporciona un marco más expresivo para formular afirmaciones sobre algunos objetos del discurso y las relaciones entre ellos.

Introducción

- En matemáticas se examinan estructuras y sus propiedades.

- Por ejemplo, **todo** número entero n es par si **existe** un número k tal que $n = 2k$.

- Los aspectos lógicos de los lenguajes naturales y artificiales presentan diferentes posibilidades expresivas.

- Por ejemplo, **todo** mamifero m es vertebrado.

Introducción

La lógica proposicional puede formalizar patrones de razonamiento simples pero es insuficiente para representar conocimiento y realizar un razonamiento práctico.

1. No permite identificar elementos que se repiten dentro de las oraciones:
   1. p : Frodo es un hobbit.
   2. q : Sam es un hobbit.
   3. r : Frodo es amigo de Sam.
2. Es limitada para tratar con ciertas partículas que tienen valor lógico, como cuantificadores e identidad:
   1. s : Todos los que hobbits que habitan en la Comarca.
   2. t : Ningún orco habita en la Comarca.
   3. u : Algunos hobbits no han salido de la Comarca.
   4. $w \wedge y$: Sauron odia a todos, incluso a sí mismo.

Introducción

- Consideremos el ejemplo: **“Frodo es un hobbit”.**
- En lógica proposicional, podríamos identificar esta afirmación con una proposición atómica p.
- No refleja la estructura lógica más fina de esta oración y qué está afirmando.
- En este contexto se trata de identificar personajes que pertenencen a cierta categoría.
- **“Frodo es amigo de Sam”** expresa una relación entre dos personajes.
- **“Todos los hobbits son de baja estatura”** expresa una propiedad de un conjunto de personajes.

Sintáxis de la Lógica de Primer Orden

Sintáxis de la lógica de primer orden

Definición

En la lógica de primer orden, la sintaxis se refiere al conjunto de reglas formales que definen cómo se pueden construir proposiciones válidas (fórmulas bien formadas) a partir de símbolos básicos. La sintaxis determina la estructura correcta de las expresiones lógicas sin considerar su significado (semántica). Esto incluye la especificación de los símbolos permitidos, cómo pueden combinarse para formar expresiones más complejas, y las reglas para el uso correcto de cuantificadores y variables.

Lenguaje de la lógica de primer orden

Definición

El alfabeto de la lógica de primer orden consiste de los siguientes componentes:

1. Dominio del discurso: $\mathbb{N}, \mathbb{Z}, \mathbb{R}, D, E, \ldots$
2. Constantes: $a, b, c, d, \ldots$
3. Funciones: $f, g, h, f_1, f_2, \ldots,$
4. Predicados: $p, q, r, p_1, p_2, hobbit, estudiante, \ldots$
5. Variables: $x, y, z, w, x_1, x_2, \ldots$
6. Conectores lógicos: $\neg, \wedge, \vee, \rightarrow, \leftrightarrow$
7. Cuantificadores: $\forall, \exists$
8. Signos de puntuación $), (,], [$

Ejemplos

Ejemplo 1:

1. Dominio: $\mathscr{N}$ los números naturales.
2. Predicados: $=$,$<$ y $>$.
3. Funciones: la función unaria sucesor definada como $s(x) = x + 1$, la funciones binarias de suma y multiplicación.
4. Constantes: 0.

Ejemplo 2:

1. Dominio: todos los seres humanos.
2. Predicados: unarios $H(x)$: x es hombre, $M(y)$: y es mujer, predicados binarios $Padre(x,y)$: x es padre de y.
3. Funciones: $Madre(x)$: retorna el término que representa la madre de x.
4. Constantes: María, Juan, Pedro, Martín, etc.

Sintáxis

Lenguaje formal de primer orden

Para referirnos a objetos, predicados y funciones de un dominio es necesario utilizar nombres. En matemáticas por ejemplo definimos $\sqrt{}$ o $\int$, en el dominio de las personas utilizamos nombres propios. Estos elementos juntos con conectores lógicos, variables y símbolos auxiliares determinan un lenguaje de primer orden y se representa mediante $\mathscr{L}$.

Componentes de un lenguaje formal

Un lenguaje de primer orden está compuesto por:

1. Vocabulario
2. Términos
3. Fórmulas

Vocabulario

Definition

Vocabulario es un conjunto de símbolos que pueden ser de dos clases, aquellos comunes a todo formalismo y aquellos particulares a cada dominio.

Vocabulario

Un vocabulario contiene:

1. Símbolos comunes a todo formalismo:
   1. Símbolos de variables: $V = \{x, y, z, x_0, y_0, z_0, ...\}$
   2. Conectores: $\neg, \wedge, \vee, \rightarrow, \longleftrightarrow$
   3. Cuantificadores: universal representado por $\forall$ y existencial representado por $\exists$.
2. Símbolos particulares de un formalismo: son constantes individuales, funciones y relaciones propias del dominio.
   1. Constantes: $C = \{a, b, c, a_0, a_1, a_2, ..., b_0, b_1, b_2, ...\}$
   2. Símbolos de función n-ádicos: $F = \{f^n, g^n, h^n, f_0^n, g_0^n, h_0^n, ...\}$
   3. Símbolos de relación n-ádicos: $P = \{P^n, Q^n, R^n, P_0^n, Q_0^n, R_0^n, ...\}$

Nota: existen muchos lenguajes de primer order diferentes dependiendo de los símbolos particulares que se incluyan.

Términos

Definition

Un término de $\mathscr{L}$ es una expresión, en la que intervienen variables constantes y símbolos de función. Los términos se utilizan para referirse a individuos específicos o elementos del dominio.

Se definen de acuerdo a las siguientes reglas:

1. Si $t \in V \cup C$ entonces t es un término, esto es toda variable (V) o constante (C) de $\mathscr{L}$ es un término de $\mathscr{L}$.
2. Si $t_1, t_2, ..., t_n$ son términos de $\mathscr{L}$ y f^n es un símbolo de función n-ádico de $\mathscr{L}$ entonces $f^n(t_1, t_2, ..., t_n)$ es un término de $\mathscr{L}$.

Nota: el conjunto de todos los términos de $\mathscr{L}$ se identifica con la letra T.

Términos

Ejemplos:

1. En el lenguaje $\mathscr{L}_N$ es posible identificar los siguientes términos:
   1. $+(2,2)$
   2. $3 * y$
2. En el lenguaje relacionados con el dominio de los seres humanos $\mathscr{L}_H$ identificamos los términos:
   1. Juan
   2. x
   3. madre(Juan)
   4. padre(madre(x))

Fórmulas Atómicas

Definition

Una fórmula atómica o predicado de $\mathscr{L}$ es una expresión en la que intervienen términos y símbolos de relación donde no existe una estructura lógica interna y corresponde a una proposición atómica en lógica proposicional.

Se define de la siguiente manera:
"Si $t_1, t_2, ..., t_n$ son términos de $\mathscr{L}$ y R^n es un símbolo de relación n-ádico de $\mathscr{L}$ entonces $R^n(t_1, t_2, ..., t_n)$ es una fórmula atómica de $\mathscr{L}$."
La palabra atómica significa indivisible y son las partes más simples que constituyen una fórmula bien formada.

Fórmulas Bien Formadas

Definition

Una **fórmula bien formada (fbf)** de $\mathscr{L}$ es una expresión, en la que intervienen fórmulas atómicas, conectores y/o cuantificadores, que pueden formarse utilizando las siguientes reglas:

1. Toda fórmula atómica de $\mathscr{L}$ es una fórmula bien formada.
2. Si A y B son fbf's de $\mathscr{L}$ entonces también lo son: $(\neg A)$,$(\neg B)$,$(A \wedge B)$,$(A \vee B)$,$(A \rightarrow B)$ y $(A \leftrightarrow B)$.
3. Si A es una fbf de $\mathscr{L}$ y $x \in V$ entonces $(\forall x)A$ y $(\exists x)A$ son fbf's.

Ejemplos

Ejemplo de predicados:

- $p(x)$: El número $x+2$ es un entero par.
- $\neg p(x)$: El número $x+2$ no es un entero par.
- $q(x,y)$: Los números $y+2$, $x-y$, $x+2y$ son enteros pares.

Para el universo de números enteros es posible realizar los siguientes reemplazos de variables:

- $p(5)$: El número 7 $(5+2)$ es un entero par. (Falso).
- $\neg p(7)$: El número 9 $(7+2)$ no es un entero par. (Verdadero)
- $q(4,2)$: Los números 4, 2, 8 son enteros pares. (Verdadero)

Cuantificadores

Cuantificador Existencial

El cuantificador existencial se expresa "Existe un x tal que ...", "Para algún x ...","Para al menos un x ..." y se representa con el símbolo $\exists$. Ejemplos $\exists x r(x)$, $\exists x, \exists y q(x,y)$ es equivalente a su forma abreviada $\exists x, y q(x,y)$

Cuantificador Universal

El cuantificador universal se expresa como "Para todo x ..." , "Para cada x ...", "Para cualquier x ...", "Para todo x,y ..." y se representa con el símbolo $\forall$.

Ocurrencia libre y ligada de una variable

Sea A una fórmula bien formada de $\mathscr{L}$ entonces se definen los siguientes conceptos.

1. Radio de acción de un cuantificador:
    1. En la fbf $(\forall x)A$ el radio de acción de $(\forall x)$ es A.
    2. En la fbf $(\exists x)A$ el radio de acción de $(\exists x)$ es A.
2. Ocurrencia ligada de una variable: una ocurrencia de la variable x en una fbf se dice que es ligada, si aparece dentro del radio de acción de un cuantificador universal $(\forall x)$ o uno existencial $(\exists x)$.
3. Ocurrencia libre de una variable: una ocurrencia de la variable x en una fbf se dice que es libre, si su aparición no es ligada.

Ejemplo

Ejemplos:

En la fórmula bien definida $(\forall x_1)(R^2(x_1, x_2) \rightarrow (\forall x_2)P^1(x_2))$ es posible comprobar:

1. $x_1$aparece ligada.
2. La primera ocurrencia de $x_2$aparece libre.
3. La segunda ocurrencia de $x_2$aparece ligada.
4. El radio de acción del cuantificador $(\forall x_2)$ es la fbf $P^1(x_2)$
5. El radio de acción de $(\forall x_1)$ es la fbf $(R^2(x_1, x_2) \rightarrow (\forall x_2)P^1(x_2))$.

Ocurrencia libre y ligada de una variable

Definition

Una fbf de $\mathscr{L}$ se dice que es una fórmula cerrada si y sólo si en ella no aparecen variables libres. En caso contrario se dice que la fórmula es abierta.

Ejemplo:

1. La fórmula $R(a, f(b))$ es cerrada ya que en ella no aparecen variables y, por lo tanto no contiene variables libres.
2. La fórmula $(\forall x)R(x, f(b))$ es cerrada, porque la única aparición de la variable x está ligada por el cuantificador universal.
3. La fórmula $R(x, f(b))$ es abierta, ya que la ocurrencia de la variable x es libre, por no aparece en el radio de acción de un cuantificador.

Instancias de primer orden a partir de una proposición

Definition

Dada una fórmula proposicional A cualquier sustitución de fórmulas de primer orden para las variables proposicionales en A produce una fórmula de primer orden denominada instancia de primer orden de A.

Ejemplo:
$((5 < x) \wedge (\neg \exists y (x = y^2)) \to (\exists y (x = y^2) \vee (5 < x))$ es una instancias de primer orden de $(p \wedge \neg q) \to (q \vee p)$.

Ejemplos de fórmulas

Ejemplos de fórmulas en $\mathscr{L}_H$:

1. "Juan es el padre de María y él la ama"

   ]

2. "Si Juan es el padre de María entonces existe alguien quien ama a María.

   ]

3. "Toda madre ama a todos sus hijos"

   ]

Ejemplos de uso de cuantificadores

Ejemplos:

- $r(x)$: "$2x$ es un número entero par". (Proposición abierta)
- $\forall x r(x)$ es una proposición verdadera. (Proposición cuantificada).
- $\exists x r(x)$ es una proposición verdadera. (Proposición cuantificada).
- $\forall x \neg r(x)$ es una proposición falsa.
- $\exists x \neg r(x)$ es una proposición falsa.

Observaciones

en una proposición abierta las variables que intervienen se denominan variables libres mientras que una proposición cuantificada se denominan variables acotadas.

Ejemplo uso de operadores lógicos

Consideremos el universo de los números reales y las siguientes proposiciones:

- $p(x) : x \geq 0$
- $q(x) : x^2 \geq 0$
- $r(x) : x^2 - 3x - 4 = 0$
- $s(x) : x^2 - 3 > 0$

Las siguientes proposiciones son verdaderas:

- $\exists x[p(x) \wedge r(x)]$ ya que $p(4) \wedge r(4)$ es verdadero.
- $\forall x[p(x) \rightarrow q(x)]$ ya que $q(x)$ nunca es falsa.
  - Para todo número real x, si $x \geq 0$, entonces $x^2 \geq 0$.
  - Todo número real no negativo tiene un cuadradado no negativo.
  - El cuadrado de cualquier número real no negativo es un número real no negativo.

Ejemplo uso de operadores lógicos

Consideremos el universo de los números reales y las siguientes proposiciones:

- $p(x) : x \geq 0$
- $q(x) : x^2 \geq 0$
- $r(x) : x^2 - 3x - 4 = 0$
- $s(x) : x^2 - 3 > 0$

Las siguientes proposiciones son falsas:

- $\forall x[q(x) \rightarrow s(x)]$ contraejemplo $q(1)$ es verdadera y $s(1)$ es falsa.
- $\forall x[r(x) \vee s(x)]$ contraejemplo $r(1)$ es falsa y $s(1)$ es falsa.

Implicación lógica

Sea $p(x)$ cualquier proposición abierta con un universo predeterminado no vacío entonces si $\forall x p(x)$ es verdadera, también lo es $\exists x p(x)$ es decir: $\forall x p(x) \Longrightarrow \exists x p(x)$ esto significa que $\forall x p(x) \to \exists x p(x)$ es una implicación logica.

Observaciones:

- $\exists x p(x)$ es verdadera siempre que $\forall x p(x)$ sea verdadera.
- Si $\exists x p(x)$ no significa que $\forall x p(x)$ sea verdadera.
- En general $\exists x p(x)$ no implica lógicamente a $\forall x p(x)$.

Ejemplo

Consideremos el universo de todos los números reales y las siguientes proposiciones:

1. Si un número es racional, entonces es un número real.
2. Si x es racional, entonces x es real.

Si $p(x) : x$ es un número racional y $q(x) : x$ es un número real entonces las expresiones anteriores informalmente expresan que $\forall x[p(x) \rightarrow q(x)]$.

Ejemplo

En el lenguaje python A es un vector de 20 valores enteros $A[1], A[2] \ldots A[20]$ donde se ejecuta el siguiente código.

For n in seq[1 to 20] do:
$A[n] = n * n - n$

Es posible observar:

1. Cada valor del vector es positivo: $\forall n(A[n] \geq 0)$.
2. El entero $A[20]$ es el valor más grande del vector: $\forall n[(1 \leq n \leq 19) \to (A[n] < A[20])]$.
3. Existen dos valores consecutivos tales que el valor mayor es el doble que el valor menor: $\exists n(A[n+1] = 2A[n])$.
4. Los valores están ordenados estríctamente en forma ascendente: $\forall n[(1 \leq n \leq 19) \to (A[n] < A[n+1])$.
5. Los valores del vector son distintos: $\forall m, n[(m < n) \to (A[m] \neq A[n])]$.

Doble implicación lógica

Definition

Sean $p(x)$,$q(x)$ proposiciones abiertas definidas en un universo dado. Las proposiciones abiertas $p(x)$,$q(x)$ son lógicamente equivalente, es decir $\forall x[p(x) \Longleftrightarrow q(x)]$ cuando $p(a) \longleftrightarrow q(a)$ es verdadera para cada reemplazo a en el universo dado.

Definition

Sean $p(x)$,$q(x)$ proposiciones abiertas definidas en un universo dado. Si la implicación $p(a) \to q(a)$ es verdadera para cada valor a del universo entonces escribimos $\forall x[p(x) \Longrightarrow q(x)]$ para denotar que $p(x)$ implica lógicamente a $q(x)$.

Contrapositiva, conversa e inversa

Sean proposiciones abiertas $p(x)$ y $q(x)$ definidas en un universo dado y la proposición cuantificada en forma universal $\forall x[p(x) \to q(x)]$ entonces se definen las siguientes formas:

Contrapositiva

La forma contrapositiva de $\forall x[p(x) \to q(x)]$ es $\forall x[\neg q(x) \to \neg p(x)]$.

Conversa

La forma recíproca de $\forall x[p(x) \to q(x)]$ es $\forall x[q(x) \to p(x)]$.

Inversa

La forma inversa de $\forall x[p(x) \to q(x)]$ es $\forall x[\neg p(x) \to \neg q(x)]$.

Ejemplo

Para el universo de los cuadriláteros del plano, sean $s(x)$ y $e(x)$ las siguientes proposiciones abiertas:

1. $s(x)$:x es un cuadrado.
2. $e(x)$:x es equilátero.

La forma positiva de la proposición $\forall x[s(x) \to e(x)]$ es verdadera.
La forma contrapositiva es $\forall x[s(x) \to e(x)] \Longleftrightarrow \forall x[\neg e(x) \to \neg s(x)]$.
La forma conversa $\forall x[e(x) \to s(x)]$ es falsa.
La forma inversa es $\forall x[e(x) \to s(x)] \Longleftrightarrow \forall x[\neg s(x) \to \neg e(x)]$.

Ejemplo

Sean las siguientes proposiciones $p(x) : |x| > 3$ y $q(x) : x > 3$ en el universo de los números reales entonces:

1. La forma positiva $\forall x[p(x) \rightarrow q(x)]$ es falsa, ej. $x = -5$.
2. La forma contrapositiva $\forall x[\neg q(x) \rightarrow \neg p(x)]$ es falsa.
3. La forma conversa $\forall x[q(x) \rightarrow p(x)]$ es verdadera.
4. La forma inversa $\forall x[\neg p(x) \rightarrow \neg q(x)]$ es verdadera.

Ejemplo

Sean las siguientes proposiciones $p(x): |x| > 3$, $q(x): x > 3$ y $r(x): x < -3$ en el universo de los números reales las siguientes proposiciones son verdaderas:

1. Proposición: $\forall x[p(x) \to (r(x) \vee q(x))]$
2. Contrapositiva: $\forall x[\neg(r(x) \vee q(x)) \to \neg p(x)]$
3. Conversa: $\forall x[(r(x) \vee q(x)) \to p(x)]$
4. Inversa: $\forall x[\neg p(x) \to \neg(r(x) \vee q(x))]$

En este caso la proposición inicial y su forma conversa son verdaderas entonces $\forall x[p(x) \longleftrightarrow (r(x) \vee q(x))]$ es verdadera y $\forall x p(x) \Longleftrightarrow \forall x[r(x) \vee q(x)]$.

Resumen

Proposición	Cuándo es verdadera?	Cuándo es falsa?
$\exists x p(x)$	Para al menos un a del universo $p(a)$ es verdadera.	Para cada a del universo $p(a)$ es falsa.
$\forall x p(x)$	Para cada reemplazo de a en el universo $p(a)$ es verdadera.	Existe al menos un reemplado de a para el cual $p(a)$ es falsa.
$\exists x \neg p(x)$	Para al menos una elección a del universo $p(a)$ es falsa de modo que $\neg p(a)$ es verdadera.	Para cada a del universo $p(a)$ es verdadera.
$\forall x \neg p(x)$	Para cada reemplazo de a del universo $p(a)$ es falsa y su negación $\neg p(a)$ es verdadera.	Existe al menos un reemplado de a para el cual $\neg p(a)$ es falsa y $p(a)$ es verdadera.

Propiedades de los cuantificadores

Sea $p(x): 2x+1=5$ y $s(x): x^2=9$ en el universo de los números enteros entonces:

- $\exists x[r(x) \wedge s(x)]$ es falsa.
- $\exists x r(x) \wedge \exists x s(x)$ es verdadera.

El cuantificador existencial no distribuye respecto al operador lógico $\wedge$ es decir: $\exists x[r(x) \wedge s(x)] \not\Leftrightarrow [\exists x r(x) \wedge \exists x s(x)]$ y que $[\exists x r(x) \wedge \exists x s(x)] \not\Rightarrow \exists x[r(x) \wedge s(x)]$.

Equivalencias e implicaciones lógicas

1. Para un universo determinado y las proposiciones abiertas $p(x)$, $q(x)$ en la variable x se cumple:
2. $\exists x[p(x) \wedge q(x)] \Longrightarrow [\exists x p(x) \wedge \exists x q(x)]$
3. $\exists x[p(x) \vee q(x)] \Longleftrightarrow [\exists x p(x) \vee \exists x q(x)]$
4. $\forall x[p(x) \wedge q(x)] \Longleftrightarrow [\forall x p(x) \wedge \forall x q(x)]$
5. $[\forall x p(x) \vee \forall x q(x)] \Longrightarrow \forall x[p(x) \vee q(x)]$
6. $\forall x \forall y p(x,y) \Longleftrightarrow \forall y \forall x p(x,y)$
7. $\exists x \exists y p(x,y) \Longleftrightarrow \exists y \exists x p(x,y)$

Equivalencias lógicas

- Sea $p(x,y): x+y=17$, la proposición $\forall x \exists y p(x,y)$ se traduce en "Para todo entero x existe un entero y tal que $x+y=17$". Esta proposición es verdad ya que para cada valor de x el valor de y se obtiene de la siguiente forma $y=x-17$.

- Ahora consideremos $\exists y \forall x p(x,y)$ que se traduce en "Existe un valor entero y tal que para todos los enteros x se cumple que $x+y=17$". Esta proposición es falsa ya que cuando se elige un valor para y sólo un valor de x satisface la ecuación.
- Generalmente $\exists y \forall x p(x,y) \not\Leftrightarrow \forall x \exists y p(x,y)$ no son lógicamente equivalentes.

Ejemplos de equivalencias lógicas

1. $\forall x[p(x) \wedge (q(x) \wedge r(x))] \iff \forall x[(p(x) \wedge q(x)) \wedge r(x)]$. Es posible demostrar la equivalencia considerando la ley asociativa.

2. $\exists x[p(x) \rightarrow q(x)] \iff \exists x[\neg p(x) \vee q(x)]$. Para cada $p(a)$ se cumple la equivalencia lógica $p(a) \rightarrow q(a) \iff \neg p(a) \vee q(a)$.

Las siguientes equivalencias también son válidas para el cuantificador existencial.

1. $\forall x \neg\neg p(x) \iff \forall x p(x)$.
2. $\forall x \neg[p(x) \wedge q(x)] \iff \forall x[\neg p(x) \vee \neg q(x)]$
3. $\forall x \neg[p(x) \vee q(x)] \iff \forall x[\neg p(x) \wedge \neg q(x)]$

Negación de cuantificadores:

1. $\neg[\forall x p(x)] \Longleftrightarrow \exists x \neg p(x)$.
2. $\neg[\exists x p(x)] \Longleftrightarrow \forall x \neg p(x)$.
3. $\neg[\forall x \neg p(x)] \Longleftrightarrow \exists x \neg\neg p(x) \Longleftrightarrow \exists x p(x)$.
4. $\neg[\exists x \neg p(x)] \Longleftrightarrow \forall x \neg\neg p(x) \Longleftrightarrow \forall x p(x)$.

Ejemplo

Sean $p(x): x$ es impar y $q(x): x^2 - 1$ es par entonces la oración "Si x es impar, entonces $x^2 - 1$ es par" se representa mediante $\forall x[p(x) \to q(x)]$ y es una proposición verdadera.

La negación de esta proposición es:
$\neg[\forall x(p(x) \to q(x))] \Longleftrightarrow \exists x[\neg(p(x) \to q(x))]$
$\Longleftrightarrow \exists x[\neg(\neg p(x) \vee q(x))] \Longleftrightarrow \exists x[\neg\neg p(x) \wedge \neg q(x)]$
$\Longleftrightarrow \exists x[p(x) \wedge \neg q(x)]$

La negación se traduce en "Existe un entero x tal que x es impar y $x^2 - 1$ es impar" lo cual es falso.

Ejemplo

Sean $r(x) : 2x + 1 = 5$ y $s(x) : x^2 = 9$ entonces $\exists x[r(x) \wedge s(x)]$ es falsa mientras que
$\neg\exists x[r(x) \wedge s(x)] \Longleftrightarrow \forall x \neg[r(x) \wedge s(x)] \Longleftrightarrow \forall x[\neg r(x) \vee \neg s(x)]$ es verdadera. La negación se traduce en “Para todo entero x $2x + 1 \neq 5$ o $x^2 \neq 9$”.

Semántica de la lógica de primer orden

Semántica de la lógica de primer orden.

Semántica de la lógica de primer orden

La semántica en la lógica de primer orden se ocupa de asignar significados a las expresiones formuladas dentro de la sintaxis de la lógica de predicados, determinando cómo estas expresiones se relacionan con los objetos y situaciones del mundo real o de modelos matemáticos abstractos. A diferencia de la sintaxis, que se enfoca en las reglas para formar expresiones válidas, la semántica se centra en interpretar estas expresiones para establecer su verdad o falsedad bajo diversas interpretaciones.

Semántica formal de la lógica de primer orden

- Las fórmulas de un lenguaje $\mathscr{L}$ expresan proposiciones de la estructura del discurso.

- El significado de las fórmulas es relativo a la estructura dada del discurso y se calcula computacionalmente en base a la estructura de la fórmula, los valores de ocurrencia de las variables y el significado de los símbolos que la componen.

- El significado preciso de una fórmula lógica está determinado por su semántica formal.

Conceptos claves de la semántica de la lógica de primer orden

- Dominio
- Interpretación
- Valoración y satisfacibilidad

Dominio

Dominio

En la lógica de primer orden, el concepto de dominio o universo de discurso juega un papel fundamental. El dominio se refiere al conjunto de todos los objetos sobre los cuales se hace referencia o se razona en un contexto lógico específico. Es el conjunto de entidades que las variables de cuantificación pueden representar o a las cuales pueden referirse.

Dominio

- **Conjunto no Vacío:** el dominio debe ser un conjunto no vacío, ya que la lógica de primer orden requiere que haya al menos un objeto sobre el cual se puedan hacer afirmaciones o negaciones.

- **Variedad de elementos:** los elementos del dominio pueden ser objetos de cualquier tipo, dependiendo del contexto específico del discurso o análisis. Por ejemplo, podrían ser números en el contexto de la aritmética, personas en un modelo sociológico, o cualquier otro objeto en distintos campos de estudio.

- **Interpretación de variables:** las variables en las expresiones de la lógica de primer orden se interpretan como elementos de este dominio. Cuando se aplica un cuantificador (universal o existencial) a una variable, se está haciendo referencia a los objetos dentro del dominio.

Dominio

- **Independencia del lenguaje:** aunque el dominio es independiente del lenguaje formal utilizado para describir la teoría, las interpretaciones de los símbolos de función, constante y predicado dentro de las fórmulas están íntimamente relacionadas con los objetos específicos del dominio.

- **Cuantificación:** los cuantificadores en la lógica de primer orden (universal "$\forall$" y existencial "$\exists$") operan sobre el dominio. Por ejemplo, una afirmación que utiliza el cuantificador universal se interpreta como verdadera si la propiedad en cuestión se cumple para todos los objetos del dominio.

Dominio

Importancia de definir claramente el dominio

El concepto de dominio es crucial porque define el alcance de lo que se puede expresar o razonar dentro de una teoría lógica. La elección del dominio afecta directamente la interpretación de las fórmulas y las conclusiones que se pueden derivar. La definición explícita del dominio es esencial para evitar ambigüedades y asegurar la precisión en el razonamiento matemático y científico.

Interpretaciones

Interpretaciones

En la lógica de primer orden, después de definir el dominio el siguiente paso es especificar la interpretación de los símbolos no lógicos utilizados en las fórmulas del lenguaje. Esto incluye la asignación de significados a símbolos de constante, símbolos de función y símbolos de predicado dentro del contexto proporcionado por el dominio establecido.

Interpretaciones

Interpretaciones

El significado de una fórmula no puede determinarse sólo a partir de los símbolos que la componen **(sintáxis)**. El significado de una fórmula es relativo a la estructura del discurso y su interpretación. Para calcular el valor de verdad de una fórmula es necesario asignar significado a los símbolos que la componen. Interpretar un formalismo consiste en seleccionar un modelo y realizar un mapeo en la estructura S.

Interpretaciones

Una interpretación I de $\mathscr{L}$ es un par (D_I, J) que consiste en:

1. Un conjunto no vacío D_I el cual es el dominio donde se realiza la interpretación I.
2. Una aplicación J que asigna:
   1. A cada símbolo de constante, a_i de $\mathscr{L}$ un elemento distinguido de D_I es decir $J(a_i) = \bar{a}_i$.
   2. A cada símbolo de función f_i^n n-ario de $\mathscr{L}$ una función $J(f_i^n) = \bar{f}_i^n$ tal que $\bar{f}_i^n : D_I^n \to D_I$.
   3. A cada símbolo de predicado R_i^n n-ario de $\mathscr{L}$ una relación $J(R_i^n) = \bar{R}_i^n$, tal que $\bar{R}_i^n \subset D_I^n$ esto es $\bar{R}_i^n = \{(d_1, d_2, \ldots, d_n) | d_i \in D_i\}$ es un conjunto de n-tuplas de D_I^n.

Nota: los elementos sobrerayados significa que se trata de una instancia del dominio.

Ejemplo

Dada la fbf $\forall x_1 \forall x_2 \exists x_3 R_1^2(g_1^2(x_1, x_3), x_2)$ en $\mathscr{L}$ con los símbolos particulares $a_1, R_1^2, f_1^1, g_1^2, g_2^2$ una posible interpretación es:

1. Asignar el conjunto de los números naturales N como dominio de la interpretación, es decir $D_I = N$.
2. Definir una función de interpretación J de la siguiente manera:
    1. A la constante a_1 se le asigna el elemento 0.
    2. Al símbolo de función f_1^1 la función sucesor ($suc : \mathrm{N} \to \mathrm{N}$), al símbolo g_1^2 la función suma ($+ : \mathrm{N}^2 \to \mathrm{N}$) y al símbolo de función g_2^2 la función producto ($x : \mathrm{N}^2 \to \mathrm{N}$).
    3. Finalmente, al símbolo de relación R_1^2 la relación de identidad ($\approx : \mathrm{N}^2 \to \{V, F\}$).

Luego la fórmula $\forall x_1 \forall x_2 \exists x_3 R_1^2(g_1^2(x_1, x_3), x_2)$ se interpreta de la siguiente forma:

$\forall x_1, x_2 \in \mathrm{N} \exists x_3 \in \mathrm{N} \mid x + x_3 \approx x_2$

Traducciones: lenguaje natural y formal

- Traducir del lenguaje formal al lenguaje natural puede ser menos complicado que su inverso.
- El lenguaje natural contiene interpretaciones que pueden variar en el tiempo y dependen de un contexto.
- El conocimiento sobre un dominio está expresado en lenguaje natural.
- Traducir de lenguaje natural a formal podría realizarse siguiendo lineamientos tales como:
  - Identificar el dominio del discurso.
  - Definir los símbolos que representarán funciones, relaciones y constantes del dominio.
- Formalizar conocimiento es un arte donde no existen reglas precisas para llevar a cabo esta tarea.

Ejemplo 1

Juan es el marido de Francisca. Juan y Francisca son el padre y madre de Guillermo, Inés y Enrique. Guillermo es padre de Arturo y Cristina. Enrique es padre de Pedro y está casado con María.

Interpretación:

1. El dominio trata sobre relaciones entre seres humanos D_H.
2. Una interpretación en el dominio $D_I$es la siguiente:
    1. $padre(x,y): x$ es el padre de y.
    2. $madre(x,y): x$ es la madre de y.
    3. $casado(x,y): x$ está casado con y.

Ejemplos

Ejemplo 2

Si consideramos los elementos del dominio: $a : Arturo$, $c : Cristina$, $e : Enrique$, $f : Francisca$, $g : Guillermo$, $i : Inés$, $p : Pedro$

Podemos formalizar las siguientes relaciones:

1. "Juan es marido de Francisca"; $casado(j, f)$
2. "Juan y Francisca son el padre y la madre de Guillermo, Inés y Enrique"; $padre(j, g) \wedge madre(f, g) \wedge padre(j, i) \wedge madre(f, i) \wedge padre(j, e) \wedge madre(f, e)$
3. "Guillermo es padre de Arturo y Cristina";$padre(g, a) \wedge padre(g, c)$
4. "Enrique es padre"; $(\exists x) padre(e, x)$

Ejemplos

Ejemplo 2

Si consideramos los elementos del dominio: $a : Arturo$, $c : Cristina$, $e : Enrique$, $f : Francisca$, $g : Guillermo$, $i : Inés$, $p : Pedro$

Podemos formalizar las siguientes relaciones:

1. "Enrique es padre"; $(\exists x) padre(e, x)$
2. "Francisca es abuela";
   $(\exists x)(\exists y)(madre(f, x) \wedge (padre(x, y) \vee madre(x, y))$
3. "Arturo es hermano de Cristina";
   $(\exists x)((padre(x, a) \vee madre(x, a)) \wedge (padre(x, c) \vee madre(x, c)))$
4. "Guillermo es el cuñado de María"; $(\exists x)(marido(g, x)(\exists y)((padre(y, x) \vee madre(y, x)) \wedge (padre(y, m) \vee madre(y, m))))$.

Ejemplo 3

Todo número natural n puede escribirse ya sea en la forma 2k o en la forma 2k+1 para algún número natural k.

Interpretación:

1. El dominio trata sobre relaciones entre números naturales D_N.
2. Una interpretación en el dominio $D_N$es la siguiente:
    1. $natural(x) : x$ es un número natural; //Símbolo de relación
    2. $x = y : x$ es igual a y;
    3. $x * y : x$ multiplicado por y; //Símbolo de función
    4. $x + y : x$ más y;
    5. a : el número 2. //Símbolo de constante

Ejemplos

Formalización versión 1

$$(\forall n)\{natural(n) \rightarrow (\exists k)[natural(k) \wedge (n = (a * k) \vee (n = ((a * k) + 1)))]\}$$

Si en lugar de considerar el dominio de los todos los números consideramos sólo los números naturales entonces podemos formalizar la definición como sigue:

Formalización versión 2

$$(\forall n)(\exists k)[n = (a * k) \vee (n = ((a * k) + 1))]$$

Valoración y Satisfacibilidad

Definition

Una valoración v, también denominada asignación en I es una aplicación que asigna a cada variable de $\mathscr{L}$ un elemento $\bar{x}$ del dominio de la interpretación D_I.

Definition

Sean A una fbf de $\mathscr{L}$, $I = (D_I, J)$ una interpretación y v una valoración en I. Una evaluación de una fórmula es una aplicación de fbf's a valores de verdad.

Valoración y Satisfacibilidad

Definition

Sea una interpretación $I = (D_I, J)$ y sea A una fbf de $\mathscr{L}$. La valoración v en I satisface la fbf A denotado por v *Sat* A si y sólo si se cumple que $Eval_{I,v}(A) = Verdadero$ caso contrario ($Falso$) decimos que v no satisface A.

Ejemplos

Ejemplo 1:
Sea la fbf $(\forall x_1)(\forall x_2)(\forall x_3)R_1^2(g_2^2(x_1,x_2),x_2)$ y la interpretación $D_I = N$ donde g_2^2 está definida por $(* : N^2 \rightarrow N)$ y R_1^2 se define como la relación de identidad $(\approx : N^2 \rightarrow \{V, F\}$ y $a_1 = 0$.
$R_1^2(g_2^2(x_1,x_2),g_2^2(x_3,x_4))$ es satisfecha por la valoración v que asigna: $v(x_1) = 2,\ v(x_2) = 6,\ v(x_3) = 3,\ v(x_4) = 4$.
En cambio la valoración $\varphi(x_1) = 2,\ \varphi(x_2) = 5,\ \varphi(x_3) = 4,\ \varphi(x_4) = 2$ no satisface la expresión.

Ejemplo 2:
$(\forall x_1)(\forall x_2)(\forall x_3)R_1^2(g_2^2(x_1,x_2),g_2^2(x_2,x_1))$ es satisfecha por cualquier valoración.

Ejemplo 3:
$(\forall x_1)R_1^2(x_1,a_1)$ no es satisfecha por valoración alguna.

Aspectos de diseño

Dos importantes aspectos a considerar :

1. Granularidad: es el conjunto de primitivas utilizadas para representar el conocimiento. En el ejemplo de las relaciones familiares se podría haber utilizado en la interpretación $esPadre(e)$ y la traduccion del enunciado habría sido inmediata.

2. Elección del dominio: al formalizar una serie de enunciados en lenguaje natural, es recomendable seleccionar una interpretación de partida cuyo universo de discurso sea el menor conjunto de entidades cuyas propiedades se deseen representar.

Ejemplos

1. Todo número es negativo o posee raíz cuadrada:
   $(\forall x)(numero(x) \to (negativo(x) \vee raiz(x))$
2. Ningún número impar es divisible por dos:
   $(\forall x)(numero(x) \wedge par(x) \to divisible(x,2))$
3. Algunos números son transcedentes:
   $(\exists x)(numero(x) \wedge transcendente(x))$
4. Algunos números no son racionales:
   $(\exists x)(numero(x) \wedge \neg racional(x))$
5. Todo entero par mayor que 4 es la suma de dos números primos:
   $(\forall z)[(par(z) \wedge z > 4) \to$
   $(\exists x)(\exists y)(sum(x,y,z) \wedge primo(x) \wedge primo(y))]$
6. Todo entero mayor que 1 es divisible por algún número primo:
   $(\forall x)[x > 1 \to (\exists y)(\exists z)(primo(y) \wedge multiplo(y,z,x))]$

Predicados Recursivos Primitivos

Introducción

Funciones numéricas:

- Por ejemplo, $f(x,y) = x^2 + 2y^x$ se puede calcular para cualesquiera valores de x e y mediante la composición de funciones.

- En el caso de la exponenciación es posible calcular el resultado de forma recursiva, si $y^x = 1$ si $x = 0$ y de otra forma $y^x = y.y^{x-1}$.

- Las funciones donde el dominio e imagen son los naturales son computables.

Función total

Input		**Multiplicación**	**Output**
x	y	$mult(x,y)$	z
2	4	$mult(2,4)$	8
5	3	$mult(5,3)$	15
$\vdots$	$\vdots$	$\vdots$	$\vdots$
7	4	$mult(7,4)$	28
Dominio $\mathbb{N}^2$		$mult : \mathbb{N}^2 \to \mathbb{N}$	Codominio $\mathbb{N}$

Función total

En matemáticas una función se dice que es total si está definida para todo elemento del Dominio.

Función parcial

Input		División	Output
x	y	$div(x, y)$	z
4	2	$div(4, 2)$	2
6	3	$div(6, 3)$	2
$\vdots$	$\vdots$	$\vdots$	$\vdots$
7	0	$div(7, 0)$	Indefinido
Dominio $\mathbb{N}^2$		$div : \mathbb{N}^2 \to \mathbb{N}$	Codominio $\mathbb{N}$

Función parcial

En matemáticas una función se dice que es parcial si existen elementos del dominio que no están asociados con algún elemento del codominio.

Funciones Recursivas Primitivas (FRP)

- Las funciones recursivas primitivas son funciones totales de la forma $f : \mathbb{N}^n \to \mathbb{N}, n \geq 0$.

- Se construyen a partir de tres funciones base (iniciales) y dos constructores:
  - Función cero
  - Función sucesor
  - Función identidad generalizada
  - Composición
  - Recursión primitiva

- Es posible aumentar poder expresivo de la recursión primitiva para obtener una clase más potente de funciones denominada funciones recursivas parciales.

Funciones Recursivas Primitivas (FRP)

Definición: Función Recursiva Primitiva

Una función f es una función recursiva primitiva sobre los números naturales $\mathbb{N}$ si f es alguna de las tres funciones bases o se obtiene a partir de otras funciones recursivas primitivas mediante la aplicación de funciones constructoras.

Funciones bases

Definición: Funciones bases

Se definen las siguientes funciones básicas de $\mathbb{N}^n \to \mathbb{N}$ de la siguiente forma:

1. **Función cero:** para cualquier $n \geq 0$, la función n-aria $cero : \mathbb{N}^n \to \{0\}$ se define como $cero_n(x_1, \ldots, x_n) = 0$ para todos los $x_1, \ldots, x_n \in \mathbb{N}$ y $n \in \mathbb{N}$. Ejemplos: $cero_3(3,7,1) = 0$, $cero_0() = 0$.
2. **Función identidad generalizada:** la función $\prod_i^n : \mathbb{N}^n \to \mathbb{N}$ donde $n > 0$ se define como $\prod_i^n(x_1, \ldots, x_n) = x_i$, es decir que devuelve el i-ésimo elemento de la n-upla $(x_1, \ldots, x_n)$ donde $1 \leq i \leq n$. Ejemplos: $\prod_2^4(3,5,9,7) = 5$, $\prod_1^1(x) = x$.
3. **Función sucesor:** la función $succ : \mathbb{N} \to \mathbb{N}$ se define como $succ(x) = x + 1$ para todo $x \in \mathbb{N}$. Ejemplo: $succ(2) = 3$.

Funciones constructoras

Definición

Una función f se puede obtener a partir de otras funciones recursivas primitivas utilizando las siguientes funciones constructoras o simplemente constructores:

- Composición
- Recursividad primitiva

Funciones constructoras

Composición

Sea $h : \mathbb{N}^m \to \mathbb{N}$ una función total, donde $m \geq 0$ y sean $g_1, \ldots, g_m$ funciones, tal que $g_i : \mathbb{N}^n \to \mathbb{N}$ donde $n \geq 0$ para $i = 1 \ldots m$. La función $f : \mathbb{N}^n \to \mathbb{N}$ se puede definir por composición a partir de las funciones h y $g_1, \ldots, g_m$ de la siguiente forma
$f(x_1, \ldots, x_n) = h(g_1(x_1, \ldots, x_n), \ldots, g_m(x_1, \ldots, x_n))$. También, se puede representar como $f(x_1, \ldots, x_n) = [h \circ (g_1, \ldots, g_m)](x_1, \ldots, x_n)$

Recursividad primitiva

Sea g una función recursiva primitiva total n-aria, y sea h una función recursiva primitiva total $n+2$-aria. Luego puede definirse una función f , $n+1$-aria, tal que para toda n–upla $(x_1, \ldots x_n) \in \mathbb{N}^n$ y $m \in \mathbb{N}$ se tiene que
$f(x_1, \ldots, x_n, 0) = g(x_1, \ldots, x_n)$ y
$f(x_1, \ldots, x_n, m+1) = h(x_1, \ldots, x_n, m, f(x_1, \ldots, x_n, m))$. En este caso se dice que f se obtiene a partir de g y h por recursión primitiva, donde g representa el *caso base* y h el *caso recursivo*.

Observaciones

- Estas funciones son **computables** sobre los números naturales, es decir que “existe un algoritmo” el cual una computadora puede ejecutar para obtener su resultado.

- Ninguna otra función que no pueda obtenerse mediante funciones base y constructores es una función recursiva primitiva.

Función constante

Función constante

Se denomina *función constante* n–aria representada por $K_j^n : \mathbb{N}^n \to \mathbb{N}$ tal que $K_j^n(x_1, \ldots, x_n) = j$ donde $n, j \in \mathbb{N}$, $n \geq 0$ y $j \geq 0$.

Ejemplos: $K_{37}^2(x_1, x_2) = 37$, $K_3^5(x_1, x_2, x_3, x_4, x_5) = 3$

Lema

Toda función constante K_j^n es recursiva primitiva.

Ejemplo 1

La función $f : \mathbb{N}^2 \to \mathbb{N}$ donde $f(x, y) = y, \forall x, y \in \mathbb{N}$ es una función recursiva primitiva ya que puede definirse como $f(x, y) = \prod_2^2(x, y)$. Ya que $f \equiv \prod_2^2$ entonces f es una función fr_{prim} recursiva primitiva.

Ejemplo 2

Ejemplo 2

La función constante $tres : \mathbb{N} \to \mathbb{N}$ tal que $tres(x) = 3, \forall x \in \mathbb{N}$ puede definirse como $tres(x) = succ(succ(succ(cero(x))))$ que formalmente es $tres(x) = [succ \circ succ \circ succ \circ cero](x)$.

Observaciones

Los ejemplos muestran un problema común a resolver, dada una función matemática f , mostrar que f es una función recursiva primitiva. ¿Cómo lograr este objetivo?

Derivación formal de una función

Derivación formal de una función

Sea f una función recursiva primitiva. Denominaremos derivación formal recursiva primitiva (DFRP) de f a una secuencia de pasos $S = [s_1, \ldots, s_k]$, tal que cada paso s_1 es una *función base* o bien una *función recursiva primitiva* que puede obtenerse a partir una o más funciones definidas en algún paso previo $s_j \in S, j < i$ por medio de los constructores de composición y recursión primitiva.

Lema

Una función f es una función recursiva primitiva si existe alguna derivación formal recursiva primitiva para f.

Ejemplo 1

Función $tres : \mathbb{N} \to \mathbb{N}$
Una derivación formal recursiva primitiva para la función *tres* puede ser la siguiente:

1. $cero(x)$ es una función fr_{pri} base.
2. $succ(x)$ es una función fr_{pri} base.
3. $[succ \circ cero](x)$ es una función fr_{pri} por composición de 1 y 2.
4. $[succ \circ succ \circ cero](x)$ es una función fr_{pri} por composición de 2 y 3.
5. $[succ \circ succ \circ succ \circ cero](x) \equiv tres(x)$ es una función fr_{pri} por composición de 4 y 2.

La secuencia de pasos S es la siguiente $S = [cero(x), succ(x), succ \circ cero(x), succ \circ succ \circ cero(x), succ \circ succ \circ succ \circ cero(x)]$

Ejemplo 2

Función $f : \mathbb{N} \to \mathbb{N}$ tal que $f(x) = x + 2$.
Una derivación formal recursiva primitiva puede ser la siguiente:

1. $succ(x)$ es una función fr_{pri} base.

2. $[succ \circ succ](x) \equiv f(x)$ es una función fr_{pri} por composición de 1.

Ejemplo 3

Función $f : \mathbb{N}^3 \to \mathbb{N}$ tal que $f(x, y, z) = x + 1$.
Una derivación formal recursiva primitiva puede ser la siguiente:

1. $\prod_1^3(x_1, x_2, x_3)$ es una función fr_{pri} base.
2. $[succ \circ \prod_1^3](x_1, x_2, x_3) \equiv f(x_1, x_2, x_3)$ es una función fr_{pri} por composición de 1.

Función suma

Función $suma : \mathbb{N}^2 \to \mathbb{N}$ tal que $suma(x_1, x_2) = x_1 + x_2$.
Es posible definir de forma recursiva la función de la siguiente forma:

$$
\begin{aligned}
suma(x_1, 0) &= \sqcap_1^1(x_1) \\
suma(x_1, x_2 + 1) &= succ(\sqcap_3^3(x_1, x_2, suma(x_1, x_2))) \\
&= succ \circ \sqcap_3^3(x_1, x_2, suma(x_1, x_2))
\end{aligned}
$$

Si $x_1 = 2$ y $x_2 = 2$, entonces:

$$
\begin{aligned}
suma(2, 1+1) = \quad & succ \circ \sqcap_3^3(2, 1, suma(2,1)) = \\
& succ \circ \sqcap_3^3(2, 1, suma(2, 0+1)) = \\
& succ \circ \sqcap_3^3(2, 1, succ \circ \sqcap_3^3(2, 0, suma(2,0))) = \\
& succ \circ \sqcap_3^3(2, 1, succ \circ \sqcap_3^3(2, 0, \sqcap_1^1(2)) = \\
& succ \circ \sqcap_3^3(2, 1, succ \circ \sqcap_3^3(2, 0, 2)) = \\
& succ \circ \sqcap_3^3(2, 1, succ(2)) = \\
& succ \circ \sqcap_3^3(2, 1, 3) = succ(3) = 4
\end{aligned}
$$

Función signo

Función *signo* $: \mathbb{N} \to \mathbb{N}$definida por:

$$signo(x) = \begin{cases} 1 & si\ x > 0 \\ 0 & en\ caso\ contrario \end{cases}$$

Es posible definir de forma recursiva la función de la siguiente forma:

$$signo(0) = cero()$$

$$signo(x+1) = succ \circ cero(x, signo(x))$$

Función sumatoria

La función $sumat : \mathbb{N} \to \mathbb{N}$ calcula la sumatoria de los números 1 a n, es decir que $sumat(x) = 0 + 1 + \cdots + x$.

Es posible definir de forma recursiva la función de la siguiente forma:

$$sumat(0) = cero()$$

$$sumat(x+1) = suma \circ (succ \circ \prod_1^2, \prod_2^2)(x, sumat(x))$$

Demostrar que $sumat(2) = 0 + 1 + 2 = 3$

Función predecesor

Función predecesor $pred : \mathbb{N} \to \mathbb{N}$definida por:

$$pred(x) = \begin{cases} 0 & si\ x = 0 \\ x - 1 & si\ x \geq 1 \end{cases}$$

Es posible definir de forma recursiva la función de la siguiente forma:

$$pred(0) = cero()$$

$$pred(x+1) = \prod_1^2 (x, pred(x))$$

Función diferencia primitiva

La diferencia primitiva $difp : \mathbb{N}^2 \to \mathbb{N}$ donde $difp(x, y) = x - y$ cuando el resultado de la diferencia sea un valor natural incluído el cero y que el resultado sea cero en caso contrario.

$$difp(x,y) = \begin{cases} 0 & si\ x < y \\ x - y & si\ x \geq y \end{cases}$$

Es posible definir de forma recursiva la función de la siguiente forma:

$$difp(x,0) = \sqcap_1^1(x)$$

$$difp(x,y+1) = (pred \circ \sqcap_3^3)(x,y,difp(x,y))$$

Función diferencia valor absoluto

La diferencia en valor absoluto $|x - y|$ es la función $difabs : \mathbb{N}^2 \rightarrow \mathbb{N}$ definida como:

$$difabs(x, y) = \begin{cases} (x - y) & si \ \ x \geq y \\ (y - x) & si \ \ x < y \end{cases}$$

Es posible definir la siguiente función recursiva primitiva de la siguiente forma:

$$difabs(x, y) = suma \circ (difp \circ (\sqcap_1^2, \sqcap_2^2), difp \circ (\sqcap_2^2, \sqcap_1^2))(x, y)$$

Función identidad

La función identidad $id : \mathbb{N} \to \mathbb{N}$ se define como $id(x) = x, \forall x \in \mathbb{N}$.
Es posible definir de forma recursiva la función de la siguiente forma:

$$id(0) = cero()$$

$$id(y+1) = (succ \circ \sqcap_2^2)(y, id(y))$$

Predicados Recursivos Primitivos

Las funciones recursivas primitivas se enriquecen mediante el modelado de relaciones matemáticas.

Predicado recursivo primitivo

Un predicado recursivo primitivo permite capturar relaciones entre números naturales mediante una función recursiva primitiva cuyo resultado es 0 o 1.

Predicados Recursivos Primitivos

Definición de predicado

Un predicado n-ario $P(\bar{n})$ es una relación n–aria sobre números naturales, esto es $P(\bar{n}) \subseteq \mathbb{N}^n$. Sea $P(\bar{n})$ un predicado n–ario, y sea $(x_1, x_2, \ldots, x_n)$ una n-upla, entonces diremos que $P(x_1, x_2, \ldots, x_n)$ es verdadero cuando $(x_1, x_2, \ldots, x_n) \in P(\bar{n})$, y diremos que $P(x_1, x_2, \ldots, x_n)$ es falso en caso contrario.

Ejemplos

$menor(x, y) = \{(x, y) \in \mathbb{N}^2 \mid x < y\}$
$igual(x, y) = \{(x, y) \in \mathbb{N}^2 \mid x = y\}$

Predicados Recursivos Primitivos

- Los predicados expresan un resultado utilizando los términos lógicos verdadero y falso.

- Estos términos pueden asociarse a dos números naturales distinguidos, por ejemplo, $1 \equiv \textit{verdadero}$ y $0 \equiv \textit{falso}$.

- Es posible asociar un predicado $P(\bar{n})$ a una función f_P que regrese 1 cuando P es verdadero y 0 cuando P es falso. Esta función se denomina *"función característica"* del predicado P.

Función características de un predicado

Definition

La función característica de un predicado $P(\bar{n})$ es una función $n-$aria $f_P : \mathbb{N} \to \{0,1\}$ definida de la siguiente forma:

$$f_P(x) = \begin{cases} 1 & \textit{si } P(x) \textit{ es verdadero} \\ 0 & \textit{si } P(x) \textit{ es falso} \end{cases}$$

Definition

Un predicado P se dice recursivo primitivo si y solo si su función característica f_P es recursiva primitiva.

Ejemplo 1

Ejemplo

Es posible demostrar que $igual(x,y)$ es un predicado recursivo primitivo si se demuestra que su función característica lo es.

- $f_{igual} : \mathbb{N}^n \to \mathbb{N}$ se define como $f_{igual} = 1 - signo(diffabs(x,y))$
- $f_{igual}(x,y) = difp \circ (K_1^2, signo \circ (difabs \circ (\prod_1^2, \prod_2^2)))(x,y)$

- $difp(x,0) = \prod_1^1(x)$, "caso base"
- $difp(x,y+1) = (pred \circ \prod_3^3)(x,y,difp(x,y))$ "caso recursivo"
- $difabs(x,y) = \begin{cases} (x-y) & si\ x \geq y \\ (y-x) & si\ x < y \end{cases}$ la cual puede expresarse como $difabs(suma(difp(x,y), difp(y,x))$

Ejemplo 2

Predicados

Los siguientes predicados son predicados recursivos primitivos.

$escero(0) = 1,$

$escero(m+1) = 0$

De forma similar, es posible definir:

$esuno(0) = 0$

$esuno(n+1) = escero(n)$

Ejemplo 3

Comparación de números enteros

El predicado menoroigual(x,y) se puede definir como $escero(difp(x,y))$.
Caso de aplicación:
$menoroigual(2,3) = escero(difp(2,3)) = escero(0) = 1$
$menoroigual(2,2) = escero(difp(2,2)) = escero(0) = 1$ y
$menoroigual(3,2) = escero(difp(3,2)) = escero(1) = 0$

Ejemplo 4

Comparación de números enteros

El predicado mayor(x,y) se puede calcular como $\mathit{difp}(\mathit{menoroigual}(x,y),1)$, es decir que:
$\mathit{mayor}(2,3) = \mathit{difp}(1,\mathit{menorigual}(2,3)) = \mathit{difp}(1,1) = 0$
$\mathit{mayor}(2,2) = \mathit{difp}(1,\mathit{menorigual}(2,2)) = \mathit{difp}(1,1) = 0$ y
$\mathit{mayor}(3,2) = \mathit{difp}(1,\mathit{menorigual}(3,2)) = \mathit{difp}(1,0) = 1$
En general la negación de un predicado recursivo primitivo es también un predicado recursivo primitivo.

Operadores lógicos

- Un predicado $P(\bar{n})$ tiene asociado un valor de verdad para una instancia particular $P(c_1, \ldots, c_n)$ donde el resultado es o bien verdadero o bien falso.
- Es posible definir nuevos predicados a partir de las operaciones lógicas de conjunción, disyunción y negación de la manera siguiente:
  - **Negación:** el predicado $\neg P(\bar{X})$ está formado por las $n-$uplas $(x_1, \ldots, x_n)$ que **no** satisfacen $P(\bar{X})$.
  - **Conjunción:** el predicado $P(\bar{X}) \wedge Q(\bar{X})$ está formado por $n-$uplas $(x_1, \ldots, x_n)$ que satisfacen $P(\bar{X})$ y $Q(\bar{X})$.
  - **Disyunción:** el predicado $P(\bar{X}) \vee Q(\bar{X})$ está formado por $n-$uplas $(x_1, \ldots, x_n)$ que satisfacen o bien $P(\bar{X})$ o bien $Q(\bar{X})$.
- Si P y Q son predicados recursivos primitivos, la definición de un nuevo predicado a través de estos operadores lógicos resulta en un predicado recursivo primitivo.

Disyunción lógica

Cálculo del operador de disyunción mediante funciones recursivas primitivas:

$$P(x,y) \vee Q(x,y) = difp(1, escero(suma(P(x,y), Q(x,y))))$$

$P(x,y)$	$Q(x,y)$	$suma(P,Q)$	$escero(suma(P,Q))$	$difp(1, escero(suma(P,Q)))$
0	0	0	1	0
0	1	1	0	1
1	0	1	0	1
1	1	2	0	1

Cálculo del operador de disyunción mediante funciones recursivas primitivas:

$$P(x,y) \wedge Q(x,y) = \mathit{difp}(1, \mathit{escero}(\mathit{mult}(P(x,y), Q(x,y))))$$

$P(x,y)$	$Q(x,y)$	$\mathit{mult}(P,Q)$	$\mathit{escero}(\mathit{mult}(P,Q))$	$\mathit{difp}(1, \mathit{escero}(\mathit{mult}(P,Q)))$
0	0	0	1	0
0	1	0	1	0
1	0	0	1	0
1	1	1	0	1

Lema

Si P y Q son predicados recursivos primitivos, entonces los predicados $\neg P$, $P \vee Q$ y $P \wedge Q$ también lo son.

Corolario

Sean P y Q predicados recursivos primitivos, y sean f_P y f_Q sus respectivas funciones características recursivas primitivas. Entonces las funciones características $f_{\neg P}$, $f_{P \vee Q}$ y $f_{P \wedge Q}$ asociadas a los predicados $\neg P$, $P \vee Q$ y $P \wedge Q$ también son recursivas primitivas.

Ejemplo 1: función signo'

Recordemos que la función $signo(x)$ está definida como:

$$signo(x) = \begin{cases} 1 & si \ x > 0 \\ 0 & en \ caso \ contrario \end{cases}$$

Es posible definir una función recursiva primitiva denominada $signo'$ de la siguiente forma:

$$signo'(x) = \begin{cases} 1 & si \ x = 0 \\ 0 & en \ caso \ contrario \end{cases}$$

Observar que $signo(x)$ devuelve siempre 1 ó 0, por lo cual puede pensarse a esta función como un predicado. Pero entonces $signo'(x) = \neg signo(x)$. Luego, por lema presentado anteriormente, puede asegurarse que $signo'$ es una función recursiva primitiva.

Predicados recursivos primitivos disjuntos

Predicados recursivos primitivos disjuntos

Si $P_1, \ldots, P_m$ son predicados recursivos primitivos, diremos que son *disjuntos* entre sí cuando para toda n–upla $(x_1, \ldots, x_n)$ se verifica que no pertenece a más de una relación determinada por los predicados $P_1, \ldots, P_m$.

Predicados recursivos primitivos disjuntos

Predicado Par

Es posible definir el predicado $par(x)$ que determina si x es un número par o no. Este predicado puede ser caculado mediante la siguiente función característica:
$par(0) = succ \circ cero()$
$par(x+1) = difp \circ (succ \circ cero, \sqcap_2^2)(x, par(x))$

Predicado Impar

Es posible definir el predicado $impar(x)$ que determina si x es un número impar o no. Este predicado puede ser caculado mediante la siguiente función característica:
$impar(x) = difp \circ (succ \circ cero, par \circ \sqcap_1^1)(x)$ es decir $[1 - par(x)]$

Ejemplo de PRPD

El predicado $par(x)$ es disjunto del predicado $impar(x)$ ya que un número natural no puede ser simultáneamente par e impar.

Función recursiva primitiva definida por casos

Definition (Función recursiva primitiva definida por casos)

Sean $P_1, \ldots, P_m$ predicados recursivos primitivos disjuntos y sean $g_1, \ldots, g_m$ funciones recursivas primitivas. Se denominará función recursiva primitiva definida por casos a toda función f con la siguiente estructura:

$$f(x_1, \ldots, x_n) = \begin{cases} g_1(x_1, \ldots, x_n) & si\ P_1(x_1, \ldots, x_n)\ es\ verdadero \\ g_2(x_1, \ldots, x_n) & si\ P_2(x_1, \ldots, x_n)\ es\ verdadero \\ \vdots & \vdots \\ g_m(x_1, \ldots, x_n) & si\ P_m(x_1, \ldots, x_n)\ es\ verdadero \end{cases}$$

Lema

Sea $g(x_1, ..., x_n)$ una función recursiva primitiva definida por casos. Entonces f es una función recursiva primitiva.

Ejemplo 2: función máximo

Es posible definir por casos la función $max : \mathbb{N}^2 \to \mathbb{N}$, donde $max(x,y)$ es el mayor valor entre x e y.

$$max(x,y) = \begin{cases} x & si\ x > y \\ y & si\ x \leq y \end{cases}$$

Formalmente:

$$max(x,y) = \begin{cases} \prod_1^2(x,y) & si\ mayor(x,y) \\ \prod_2^2(x,y) & si\ \neg mayor(x,y) \end{cases}$$

Donde $mayor(x,y)$ corresponde a un predicado que tiene asociado el valor verdadero cuando $x > y$ y falso en caso contrario. La definición $max(x,y)$ es una función por casos, y como tal (por lema) es una función recursiva primitiva.

Ejemplo 3: función máximo'

Considerando la función *signo* es posible obtener una función característica para el predicado $mayor(x, y)$ mediante las siguientes funciones mutuamente excluyentes:

$$f_{mayor}(x, y) = signo(x - y)$$

$$f_{menor\ o\ igual}(x, y) = signo'(x - y)$$

$$max(x, y) = f_{mayor}(x, y) * x + f_{menor\ o\ igual}(x, y) * y$$

Formalmente:

$max(x, y) =$
$suma(producto(\prod_1^2(x, y), signo(difp(x, y))), producto(\prod_2^2(x, y), signo'(difp(x, y))))$

De donde se sigue:

$max(x, y) = suma \circ (producto \circ (\prod_1^2, signo \circ difp \circ (\prod_1^2, \prod_2^2)), producto \circ (\prod_2^2, signo' \circ difp \circ (\prod_1^2, \prod_2^2)))(x, y)$

Importante

Dada una cierta función f, no existe una única forma de definir f a través de funciones recursivas primitivas, es decir que pueden existir distintas derivaciones formales para una misma función f .

Fin

$\boxed{0}$_home_lwittgenstein_Dropbox_data