Inducción
Matemática Estructural y Lógica

Miguel De Ávila

17 de abril de 2018

Inducción

El axioma 5 de Peano establece la siguiente propiedad acerca de los números naturales:

$$(\forall A : \mathbb{P}(\mathbb{N}) | 0 \in A \wedge (\forall n : A | : S(n) \in A) : A = \mathbb{N})$$

Dicho de otro modo, si quisieramos probar que un conjunto $A \subseteq \mathbb{N}$ es de hecho todos los naturales, debemos probar dos cosas:

- $0 \in A$
- Para todo $n \geq 0$, si $n \in A$, entonces $(n+1) \in A$

Esquema de inducción

Podemos reescribir el axioma para que hable sobre propiedades:

Teorema

Esquema de inducción Sea $P : \mathbb{N} \rightarrow bool$ *un predicado sobre números naturales. Entonces, si se tiene:*

- $P(0)$,
- $P(k) \Rightarrow P(k+1)$;

entonces P *es verdadero para todos los naturales, es decir:* $(\forall n : \mathbb{N} | : P(n))$.

¿Cómo probar cosas por inducción?

Suponga que se desea mostrar que $(\forall n : \mathbb{N}| : P(n))$. Para una prueba por inducción:

- Definir P si aún no se ha definido.
- Mostrar que $P(0)$ es cierto. Esto se llama *probar el caso base*.
- Suponer que $P(k)$ es verdadero para un $k \geq 0$ arbitrario. $P(k)$ se conoce como la *hipótesis de inducción*.
- A partir de la hipótesis de inducción, mostrar que $P(k+1)$ vale.

Como conclusión, se tiene que $P(n)$ vale para todos los naturales.

Ejemplos

Se puede mostrar por inducción:

Teorema

- $(+k|0 \leq k \leq n : k) = \frac{n(n+1)}{2}$.
- *Para todo* $n \in \mathbb{N}$, $n(n+1)$ *es divisible por 2.*
- *Para todo* $n \in \mathbb{N}$ *se tiene que* $3|(n^3 - n)$.
- $(+k|1 \leq k \leq n : k^2) = \frac{n(n+1)(2n+1)}{6}$.

Demostraciones erradas con inducción

Se puede llegar a resultados absurdos si no se utiliza bien el predicado de inducción, por ejemplo:

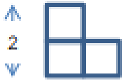
Teorema

(Teorema errado) Todos los calcetines en el mundo son del mismo color.

La demostración se hace por inducción: Se "demuestra" que en todo conjunto de n calcetines, todos los calcetines deben ser del mismo color.

Demostración gráfica por inducción

Suponga que se tiene un patio de dimensiones $2^n \times 2^n$. En uno de los cuadros se coloca una estatua, y el resto de los cuadros debe ser embaldosados, pero las únicas baldosas disponibles son en forma de L:



Teorema

Es posible embaldosar el patio con solo baldosas en forma de L.

Inducción corrida

Con el esquema de inducción podemos, en principio, demostrar solo propiedades para todos los números naturales. Nos gustaría poder demostrar teoremas de números naturales (o enteros!) que solo son ciertos a partir de cierto punto.

Teorema

(Inducción corrida) Sea z_0 un entero y P un predicado sobre los enteros. Suponga que se tiene lo siguiente:

- $P(z_0)$.
- $P(z) \Rightarrow P(z+1)$, *para todo* $z \geq z_0$.

Entonces se tiene $(\forall z : \mathbb{Z} | z \geq z_0 : P(z))$.

Inducción fuerte

Además de esto último, podemos fortalecer la hipótesis de inducción: en vez de suponer que P vale para el paso anterior, podemos suponer que vale para todos los pasos anteriores. Enunciamos el siguiente teorema:

Teorema

(Inducción fuerte) Sea $P : \mathbb{Z} \rightarrow$ *bool un predicado sobre los enteros y* z_0 *un entero cualquiera. Suponga que se tiene lo siguiente:*

- $P(z_0)$.
- $(\forall z | z_0 \leq z < k : P(z)) \Rightarrow P(k)$ *para todo* $k \geq z_0$.

Entonces se tiene $(\forall z : \mathbb{Z} | z \geq z_0 : P(z))$

Teoremas con inducción fuerte

Teorema

Cualquier cantidad superior a 3 centavos se puede representar con monedas de 2 y 5 centavos.

Teorema

Todo número $n > 1$ *se puede descomponer únicamente en sus factores primos.*

Teoremas con inducción fuerte

Suponga un juego con dos jugadores y una bolsa que tiene n bolas. Cada jugador en su turno puede sacar 1, 2 o 3 bolas. El jugador que saca la última bola pierde.

Teorema

El jugador 1 gana si $n =_4 0$, $n =_4 2$ o $n =_4 3$. Si $n =_4 1$ entonces gana el jugador 2.

Definiciones recursivas

Es usual encontrar objetos matemáticos que se definen recursivamente. Un ejemplo sencillo es el factorial, que es una función de $\mathbb{N}$ en $\mathbb{N}$:

$$
\begin{aligned}
0! &= 1 \\
n! &= n(n-1)!
\end{aligned}
$$

El factorial de 0 está bien definido. Y si el factorial de $(n-1)$ está bien definido, entonces el factorial de n también.
La construcción inductiva nos garantiza que el factorial de cualquier número natural está bien definido.

Inducción en otras estructuras

- Los números naturales son por excelencia la estructura para definir funciones y probar propiedades recursivas utilizando inducción.
- Sin embargo, se puede hacer inducción sobre muchas más estructuras.
- Estas estructuras las llamaremos *bien fundadas*

Definición

Una estructura con noción de orden se llama bien fundada si no tiene cadenas infinitas descendientes.

Estas son las estructuras sobre las que aplicaremos inducción estructural.

Listas como estructuras

Podemos definir en principio listas de números naturales. Como por ejemplo:

$$< 1, 3, 2, 5, 1, 11, 20 >$$

Si quisieramos definir cualquier lista finita posible, podríamos hacer una definición inductiva:

- *vac* es una lista y corresponde a la lista vacía.
- $ins : Lista \times \mathbb{N} \to Lista$ es una función que inserta un elemento al final de una lista ya existente. Si l es una lista y $n \in \mathbb{N}$ entonces $ins(l, n)$ también es una lista.

Funciones definidas sobre listas

Podemos definir recursivamente funciones sobre listas. Por ejemplo, la función $tam : Lista \to \mathbb{N}$

$$\begin{aligned} tam(vac) &= 0 \\ tam(ins(l, x)) &= tam(l) + 1 \end{aligned}$$

O la función $concat : Lista \times Lista \to Lista$

$$\begin{aligned} concat(l, vac) &= l \\ concat(l, ins(l', x)) &= ins(concat(l, l'), x) \end{aligned}$$

Inducción estructural

Para probar que un cierto predicado P se cumple para todos los elementos con una cierta estructura:

1. Se demuestra que P se cumple para el o los casos base.
2. Se toma un elemento x que se construyó inductivamente a partir de $x_1, \ldots, x_n$. Se demuestra que x cumple con P, usando como hipótesis que $x_1, \ldots, x_n$ cumplen con P. Aquí, $x_1, \ldots, x_n$ son objetos más sencillos que x, mientras que x es el objeto más complejo.

Una estructura puede tener varias maneras recursivas de construir nuevos elementos. El segundo paso debe repetirse una vez por cada una de estas.

Teoremas sobre listas

Podríamos probar por inducción el siguiente teorema sobre listas:

Teorema

$tam(concat(l_1, l_2)) = tam(l_1) + tam(l_2)$

Más funciones sobre listas

Primer elemento (función parcial):

$$\begin{aligned} prim(ins(l, x)) &= prim(l) \text{ si } l \neq vac \\ prim(ins(vac, x)) &= x \end{aligned}$$

Último elemento:

$$ult(ins(l, x)) = x$$

Mínimo (para listas de naturales):

$$\begin{aligned} min(vac) &= \infty \\ min(ins(l, x)) &= min(min(l), x) \end{aligned}$$

Teoremas sobre listas

Teorema

1. $min(concat(l_1, l_2)) = min(min(l_1), min(l_2))$
2. $rev(rev(l)) = l.$
3. $rev(concat(l_1, l_2)) = concat(rev(l_1), rev(l_2)).$
4. $tam(rev(l)) = tam(l).$
5. $min(rev(l)) = min(l).$

Árboles binarios

Los árboles binarios sobre un conjunto X pueden ser definidos recursivamente así:

- *arvac* es un árbol binario y representa el árbol binario vacío.
- La función $cons : Arbin \times X \times Arbin \Rightarrow Arbin$ es la función que construye nuevos árboles. Si a_1, a_2 son árboles binarios y $x \in X$ entonces $cons(a_1, x, a_2)$ también es un árbol binario.

Podemos definir raíz, hijo izquierdo, hijo derecho, tamaño y altura de un árbol binario.

Funciones sobre árboles binarios

Altura:

$$
\begin{aligned}
alt(arvac) &= 0 \\
alt(cons(a_1, x, a_2)) &= max(alt(a_1), alt(a_2)) + 1.
\end{aligned}
$$

Tamaño:

$$
\begin{aligned}
tam(arvac) &= 0 \\
tam(cons(a_1, x, a_2)) &= 1 + tam(a_1) + tam(a_2).
\end{aligned}
$$

Raíz:

$$raiz(cons(a_1, x, a_2)) = x$$

Teoremas sobre árboles binarios

Podríamos probar el siguiente teorema usando inducción estructural:

Teorema

$tam(a) \leq 2^{alt(a)} - 1$

Fórmulas bien formadas

Definimos las fórmulas bien formadas (fbf) de la lógica proposicional de forma inductiva:

- Cualquier letra proposicional es una fbf.
- Si α es una fbf, entonces $\neg(\alpha)$ es una fbf.
- Si α y β son fbf entonces $(\alpha) \star (\beta)$ es una fbf, donde $\star$ es un operador binario.

¿Qué operaciones y funciones podemos definir sobre las fórmulas bien formadas?

Teoremas de fbf

Para una fórmula bien formada podemos demostrar:

Teorema

Toda ocurrencia de un paréntesis izquierdo está precedida por un operador binario o una negación.

Teorema

Para toda fbf α, $P(\alpha) = 2N(\alpha) + 4B(\alpha)$, *donde* N *representa el número de negaciones,* B *el número de operadores binarios y* P *el número de paréntesis.*