Práctico 1
Inducción y recursión

Observación general Para cada lenguaje (conjunto) definido inductivamente a lo largo de este práctico, debe poder explicitar su principio de inducción y su esquema de recursión primitiva. Además, para cada función que siga el esquema de recursión primitiva, debe poder probar que efectivamente lo sigue, explicitando el uso de dicho esquema. Realice tantas de estas tareas como sean necesarias para obtener un buen manejo de los mismos.

Recordemos que para probar que una definición sigue el esquema de recursión primitiva, por ejemplo el de los naturales, debe proporcionar la constante $f_0 \in \mathbb{N}$ y la función $f_s : \mathbb{N} \to \mathbb{N}$ exigidos por el esquema.

Ejercicio 1 (Reglas sobre reales)

Considere el conjunto $\mathbb{R}$ de los números reales; la lista de subconjuntos de $\mathbb{R}$ de la columna izquierda; y los conjuntos de reglas de la columna derecha. Indique cuáles subconjuntos satisfacen cuáles conjuntos de reglas.

A. $\mathbb{N}$ B. $\mathbb{Z}$	1. I $2 \in S$ II Si $n \in S$, entonces $n+2 \in S$
C. $\{2n+1 : n \in \mathbb{N}\}$ D. $\{\pi + k : k \in \mathbb{Z}\}$ E. $\emptyset$	2. I $3 \in S$ II Si $n \in S$, entonces $n+1 \in S$ III Si $n \in S$, entonces $n-1 \in S$
F. $\mathbb{R}$	3. I Si $n+1 \in S$, entonces $n \in S$

Por ejemplo, para investigar si el conjunto A cumple el conjunto de reglas 1 es necesario preguntarse si se cumplen las siguientes reglas:

I $2 \in \mathbb{N}$

II Si $n \in \mathbb{N}$, entonces $n+2 \in \mathbb{N}$

Ejercicio 2 (Inducción sobre los naturales)

Enuncie el principio de inducción primitiva para el conjunto `Par` definido inductivamente por las siguientes cláusulas:

I $0 \in$ `Par`

II Si $n \in$ `Par` entonces $n+2 \in$ `Par`

Pruebe utilizando este principio, que para todo $n \in$ `Par`, existe $m \in \mathbb{N}$ tal que $n = m + m$.

Ejercicio 3 (Inducción sobre los naturales)

Considere el conjunto $\mathbb{N} \times \mathbb{N}$ de las parejas de naturales, y el conjunto inductivo F definido mediante las siguientes reglas:

I $\langle 0, 1\rangle \in F$

II Si $\langle n, m\rangle \in F$, entonces $\langle m, n+m\rangle \in F$

Use el principio de inducción de los naturales para demostrar que

$$(\bar{\forall} n \in \mathbb{N})\, \langle \texttt{Fibo}(n), \texttt{Fibo}(n+1) \rangle \in F$$

donde la función `Fibo` se define como:

$$\begin{array}{ll} \texttt{Fibo} : \mathbb{N} \to \mathbb{N} & \\ \texttt{Fibo}(0) & = 0 \\ \texttt{Fibo}(1) & = 1 \\ \texttt{Fibo}(n+2) & = \texttt{Fibo}(n) + \texttt{Fibo}(n+1) \end{array}$$

Ejercicio 4 (Reglas sobre palabras)

Considere el alfabeto Σ formado por las letras a y b; la lista de subconjuntos de Σ^* de la columna izquierda; y los conjuntos de reglas de la columna derecha. Indique cuáles subconjuntos satisfacen cuáles conjuntos de reglas.

A. Σ B. $\{w \in \Sigma^* : w \text{ tiene largo par}\}$ C. $\{w \in \Sigma^* : w \text{ termina en } a\}$ D. $\{w \in \Sigma^* : w \text{ es un palíndrome}\}$ E. $\emptyset$ F. Σ^*	1. I $\varepsilon \in S$ II Si $w \in S$, entonces $awa \in S$ 2. I $a \in S$ II $b \in S$ III Si $w \in S$, entonces $ww \in S$ 3. I Si $w \in S$, entonces $wa \in S$

Ejercicio 5 (Principio de inducción primitiva)

Considere los conjuntos de reglas indicados en Ejercicio 1 y Ejercicio 4. Indique qué conjuntos (lenguajes) quedan definidos inductivamente por ellos. Para aquellos que no sean vacíos, proporciones elementos que le pertenezcan y elementos que no le pertenezcan. Enuncie y demuestre el principio de inducción primitiva para cada uno de ellos.

Ejercicio 6 (Lenguajes)

Considere el alfabeto $\Sigma = \{a, b, c\}$, y los lenguajes Δ y Γ definidos inductivamente con las siguientes reglas.

I $\varepsilon \in \Gamma$ II $a \in \Gamma$ III Si $\alpha \in \Gamma$ y $\beta \in \Gamma$, entonces $b\alpha c\beta b \in \Gamma$	I $\varepsilon \in \Delta$ II Si $\alpha \in \Delta$, entonces $b\alpha bc \in \Delta$ III Si $\alpha \in \Delta$, entonces $b\alpha ba \in \Delta$

a. Encuentre palabras de Σ^* que no pertenezcan a Γ. Análogo para Δ.
b. Muestre que Γ no está incluído en Δ y que Δ tampoco está incluído en Γ.

Para probar que un lenguaje no está incluído en otro debe proporcionar una palabra que pertenezca al primer lenguaje y no pertenezca al segundo, con las justificaciones que correspondan.

Ejercicio 7 (Lenguajes)

Considere el alfabeto $\Sigma = \{a, b, c\}$, y los lenguajes Δ y Γ definidos en el Ejercicio 6. Demuestre, usando el principio de inducción que corresponda, que

a. el largo de las palabras de Δ es múltiplo de tres
b. todas las palabras de Δ tiene una cantidad par de ocurrencias de la letra b
c. todas las palabras de Γ tiene una cantidad par de ocurrencias de la letra b
d. en Γ no hay palabras de largo dos
e. en Γ hay palabras de cualquier largo salvo dos

Ejercicio 8 (Definiciones inductivas)

Cada ítem describe un lenguaje sobre el alfabeto $\Sigma = \{a, b\}$. Se le pide la definición inductiva de cada lenguaje.

L1 = $\{\varepsilon, a, aa, aaa, aaaa, \ldots\}$

L2 = $\{\varepsilon, ab, aabb, aaabbb, aaaabbbb, \ldots\}$

L3 = $\{b, aba, aabaa, aaabaaa, aaaabaaaa, \ldots\}$

L4 = $\{\varepsilon, ab, abab, ababab, abababab, \ldots, ba, baba, bababa, babababa, \ldots\}$

L5 = $\{\varepsilon, a, ab, aba, abab, ababa, ababab, \ldots\}$

L6 = $\{\alpha \in \Sigma^* : \alpha \text{ es un palíndrome}\}$

Ejercicio 9 (Inserción a derecha)

Considere el alfabeto $\Sigma = \{a, b\}$, y los lenguajes Δ y Γ definidos inductivamente con las siguientes reglas. Demuestre que $\Gamma = \Delta$.

I $\varepsilon \in \Gamma$ II Si $\alpha \in \Gamma$, entonces $a\alpha \in \Gamma$ III Si $\alpha \in \Gamma$, entonces $b\alpha \in \Gamma$	I $\varepsilon \in \Delta$ II Si $\alpha \in \Delta$, entonces $\alpha a \in \Delta$ III Si $\alpha \in \Delta$, entonces $\alpha b \in \Delta$

Ejercicio 10 (Relaciones inductivas)

Considere la siguiente definición inductiva de la relación $S \subseteq \mathbb{N} \times \mathbb{N}$:

I Si $n \in \mathbb{N}$ entonces $\langle n, n\rangle \in S$
II Si $\langle n, m\rangle \in S$ entonces $\langle n, m+1\rangle \in S$

a. Indique cuáles de las siguientes afirmaciones son correctas y justifique su respuesta usando la definición de S.

$$\langle 0, 0\rangle \in S \qquad 0 \in S \qquad \langle \pi, \pi\rangle \in S \qquad \langle 2, 3\rangle \in S \qquad \langle 3, 2\rangle \in S$$

b. Enuncie el principio de inducción primitiva para S.

c. Dé una definición por comprensión del conjunto S.

d. Considere la siguiente definición inductiva de la relación $Q \subseteq \mathbb{N} \times \mathbb{N}$:

  I Si $n \in \mathbb{N}$ entonces $\langle 0, n \rangle \in Q$
  II Si $\langle n, m \rangle \in Q$ entonces $\langle n+1, m+1 \rangle \in Q$

  ¿Cuáles de las siguientes afirmaciones son correctas? Justifique su respuesta.

$$S \subset Q \qquad Q \subset S \qquad S \subseteq Q \qquad Q \subseteq S \qquad Q = S$$

Ejercicio 11 (Recursión en los naturales)

Considere la siguiente función definida por recursión primitiva en $\mathbb{N}$:

$$\begin{array}{ll} f : \mathbb{N} \to \mathbb{N} & \\ f(0) & = 0 \\ f(n+1) & = f(n) + 2 \times (n+1) \end{array}$$

Pruebe que $(\bar{\forall} n \in \mathbb{N}) f(n) = n \times (n+1)$.

Ejercicio 12 (Parejas de naturales)

El conjunto $\mathbb{N} \times \mathbb{N}$ de las parejas de naturales puede definirse inductivamente con distintos conjuntos de reglas. Por ejemplo,

I Si $n \in \mathbb{N}$, entonces $\langle 0, n \rangle \in \mathbb{N} \times \mathbb{N}$ II Si $n \in \mathbb{N}$, entonces $\langle n+1, 0 \rangle \in \mathbb{N} \times \mathbb{N}$ III Si $\langle n, m \rangle \in \mathbb{N} \times \mathbb{N}$, entonces $\langle n+1, m+1 \rangle \in \mathbb{N} \times \mathbb{N}$	I Si $n \in \mathbb{N}$, entonces $\langle n, 0 \rangle \in \mathbb{N} \times \mathbb{N}$ II Si $\langle n, m \rangle \in \mathbb{N} \times \mathbb{N}$, entonces $\langle n, m+1 \rangle \in \mathbb{N} \times \mathbb{N}$

a. Identifique las siguientes funciones, o sea, señale qué función conocida es computada por las siguientes ecuaciones. Tome en cuenta que la diferencia en los naturales cumple $0 - 1 = 0$.

$$\begin{array}{ll} f : \mathbb{N} \times \mathbb{N} \to \mathbb{N} & \\ f(\langle 0, n \rangle) & = 0 \\ f(\langle n+1, 0 \rangle) & = 0 \\ f(\langle n+1, m+1 \rangle) & = 1 + f(\langle n, m \rangle) \end{array} \qquad \begin{array}{ll} g : \mathbb{N} \times \mathbb{N} \to \mathbb{N} & \\ g(\langle n, 0 \rangle) & = n \\ g(\langle n, m+1 \rangle) & = g(\langle n, m \rangle) - 1 \end{array}$$

b. Pruebe mediante inducción que

$$\begin{array}{ll} (\bar{\forall} n \in \mathbb{N}) & f(\langle n, 0 \rangle) = 0 \\ (\bar{\forall} \langle n, m \rangle \in \mathbb{N} \times \mathbb{N}) & g(\langle n, m \rangle) = g(\langle n+1, m+1 \rangle) \end{array}$$

c. Redefina f y g usando el otro esquema de recursión primitiva. Revise las pruebas inductivas del ítem anterior.

Ejercicio 13 (Concatenación)

El conjunto $\Sigma^* \times \Sigma^*$ puede definirse inductivamente con las siguientes reglas:

  I Si $w \in \Sigma^*$, entonces $\langle \varepsilon, w \rangle \in \Sigma^* \times \Sigma^*$

II Si $\langle u, w\rangle \in \Sigma^* \times \Sigma^*$ y $a \in \Sigma$, entonces $\langle au, w\rangle \in \Sigma^* \times \Sigma^*$

Considere la siguiente definición de concatenación:

$$\begin{array}{l} \texttt{append} : \Sigma^* \times \Sigma^* \to \Sigma^* \\ \texttt{append}(\langle \varepsilon, w\rangle) \quad = w \\ \texttt{append}(\langle au, w\rangle) = a\texttt{append}(\langle u, w\rangle) \end{array}$$

Demuestre que ε es neutro de esta function, es decir:

$$(\bar{\forall} u \in \Sigma^*)\texttt{append}(\langle \varepsilon, u\rangle) = u \text{ y } (\bar{\forall} u \in \Sigma^*)\texttt{append}(\langle u, \varepsilon\rangle) = u$$

Ejercicio 14 (Funciones sobre tiras)

Considere un alfabeto Σ arbitrario. Defina usando el esquema de recursión primitiva adecuado las siguientes funciones.

$$\begin{array}{r} \texttt{largo} : \Sigma^* \to \mathbb{N} \\ \texttt{duplicar} : \Sigma^* \to \Sigma^* \\ \texttt{invertir} : \Sigma^* \to \Sigma^* \end{array} \qquad \begin{array}{r} \texttt{largo} : \Sigma^+ \to \mathbb{N} \\ \texttt{último} : \Sigma^+ \to \Sigma \\ \texttt{principio} : \Sigma^+ \to \Sigma^* \\ \texttt{primero} : \Sigma^+ \to \Sigma \end{array} \qquad \begin{array}{r} \texttt{resto} : \Sigma^+ \to \Sigma^* \\ \texttt{duplicar} : \Sigma^+ \to \Sigma^+ \\ \texttt{invertir} : \Sigma^+ \to \Sigma^+ \end{array}$$

Demuestre inductivamente que el largo de una palabra duplicada es el doble de la palabra original.

Demuestre que el largo de concatenar dos tiras es la suma de sus largos.

Observe que el dominio de las funciones de las columnas a la derecha no considera la palabra ε. Las funciones cuya aplicación a la palabra vacía no tienen sentido[1] tienen como dominio a Σ^+.

Los siguientes ejemplos aclaran el significado de las funciones pedidas.

$$\begin{array}{r} \texttt{largo}(abcdc) = 5 \\ \texttt{resto}(abcdc) = bcdc \\ \texttt{duplicar}(abc) = aabbcc \end{array} \qquad \begin{array}{r} \texttt{primero}(abcdc) = a \\ \texttt{último}(abcdc) = c \end{array} \qquad \begin{array}{r} \texttt{principio}(abcdc) = abcd \\ \texttt{invertir}(abcdc) = cdcba \end{array}$$

Ejercicio 15 (Más funciones sobre tiras)

a. Defina inductivamente el conjunto $\Sigma^* \times \Sigma$.

b. Identifique la siguiente función.

$$\begin{array}{l} f : \Sigma^* \times \Sigma \to \{0, 1\} \\ f(\langle \varepsilon, x\rangle) \quad = 0 \\ f(\langle xw, y\rangle) = \text{si } x = y \text{ entonces } 1 \text{ sino } f(\langle w, y\rangle) \end{array}$$

c. Defina usando el esquema de recursión primitiva adecuado las siguientes funciones.

$$\begin{array}{r} \texttt{cant} : \Sigma^* \times \Sigma \to \mathbb{N} \\ \texttt{copiar} : \Sigma^* \times \mathbb{N} \to \Sigma^* \\ \texttt{sacar\_de\_la\_izquierda} : \Sigma^* \times \mathbb{N} \to \Sigma^* \\ \texttt{primera\_posición} : \Sigma^* \times \Sigma \to \mathbb{N} \end{array}$$

[1] En algunos casos, podemos totalizar la función: por ejemplo, podemos definir $\texttt{principio}(\varepsilon) := \varepsilon$. Pero en ocasiones esto no es posible: ¿Cuál sería la primera letra de ε? ¿Y la última?

Los siguientes ejemplos aclaran el significado de las funciones pedidas.

$$\begin{array}{rr} \texttt{cant}(\langle abcdc, c\rangle) = 2 & \texttt{copiar}(\langle ab, 3\rangle) = ababab \\ \texttt{sacar\_de\_la\_izquierda}(\langle abcdc, 3\rangle) = dc & \texttt{primera\_posición}(\langle abcdc, c\rangle) = 3 \\ \texttt{sacar\_de\_la\_izquierda}(\langle abcdc, 7\rangle) = \varepsilon & \texttt{primera\_posición}(\langle abcdc, e\rangle) = 6 \end{array}$$

Ejercicio 16 (Orden)

Considere un alfabeto Σ enriquecido con una relación de orden $\leq$.

a. Defina la función `entre` : $\Sigma^* \times \Sigma \times \Sigma \to \{0, 1\}$ que devuelve 1 si y solamente si todas las letras de su primer argumento están entre las letras dadas. Por ejemplo, considerando las letras del alfabeto castellano con el orden usual, `entre`$(\langle \text{jose}, b, t\rangle) = 1$ y `entre`$(\langle \text{mama}, b, z\rangle) = 0$.

b. Defina la función `insord` : $\Sigma^* \times \Sigma \to \Sigma^*$ que inserta ordenadamente una nueva letra en una tira ya ordenada. Por ejemplo, `insord`$(\langle abcde, d\rangle) = abcdde$. Indique qué hace su función si la lista de entrada se encuentra desordenada.

Ejercicio 17 (Reemplazo)

Consideremos una función $f : \Sigma \to \Sigma^*$. Decimos que w se obtiene por reemplazo a partir de u usando f, cuando w es el resultado de reemplazar cada letra de u de acuerdo a la función f.

a. Defina la función `reemplazo` : $\Sigma^* \times (\Sigma \to \Sigma^*) \to \Sigma^*$. Esa función debe cumplir que, si $\Sigma = \{a, b\}$ y f es tal que $f(a) = aba$ y $f(b) = bba$, entonces `reemplazo`$(\langle abab, f\rangle) = ababbaababba$.

b. Encuentre una función f adecuada para poder probar la siguiente propiedad, con `duplicar` definida en Ejercicio 14:

$$(\bar{\forall} w \in \Sigma^*)\texttt{duplicar}(w) = \texttt{reemplazo}(\langle w, f\rangle)$$

Ejercicio 18 (Recursión sobre un componente de un par)

Se quiere definir una función `últ_pos` que cumple

$$\texttt{últ\_pos}(\langle abcdc, c\rangle) = 5 \qquad \texttt{últ\_pos}(\langle abcdc, e\rangle) = 0$$

Considere la siguiente definición:

$$\begin{array}{ll} \texttt{últ\_pos} : \Sigma^* \times \Sigma \to \mathbb{N} & \\ \texttt{últ\_pos}(\langle \varepsilon, x\rangle) & = 0 \\ \texttt{últ\_pos}(\langle yw, x\rangle) & = \text{ si } \texttt{últ\_pos}(\langle w, x\rangle) = 0 \text{ y } x \neq y \text{ entonces } 0 \text{ sino } 1 + \texttt{últ\_pos}(\langle w, x\rangle) \end{array}$$

Demuestre que[2]

$$(\bar{\forall} w \in \Sigma^*)(\bar{\forall} a \in \Sigma)\texttt{últ\_pos}(\langle w, a\rangle) = 0 \qquad \Leftrightarrow \texttt{in}(\langle w, a\rangle) = 0$$

Ejercicio 19 (Numerales binarios)

a. Considere $\Sigma = \{0, 1\}$. Llamamos numerales binarios a las palabras de Σ^+. Defina inductivamente el lenguaje B de los numerales binarios.

b. Defina recursivamente la función `eval` : $B \to \mathbb{N}$ que devuelve el natural representado por un numeral binario. Es decir, `eval`$(10100) = 20$ y `eval`$(000100) = 4$. Para resolver este problema, recuerde que el binario 10100 representa $1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 0 \times 2^0$ (O lo que es lo mismo por la regla de Horner, $(((1 \times 2 + 0) \times 2 + 1) \times 2 + 0) \times 2 + 0)$.

[2] Donde la función `in` es la que se pedía identificar en Ejercicio 15.

c. Defina recursivamente la función `par` $: B \to \{0,1\}$ que devuelve 1 si la cantidad de unos en el numeral es par, y 0 si es impar.

Ejercicio 20 (Árboles)

Considere la siguiente definición inductiva de árboles binarios, donde $\Sigma = B \cup \{\oplus, \otimes,), (\}$, y B es el definido en Ejercicio 19.

I Si $b \in B$, entonces $b \in T$

II Si $x \in T$ y $z \in T$, entonces $(x \oplus z) \in T$

III Si $x \in T$ y $z \in T$, entonces $(x \otimes z) \in T$

a. Defina las funciones `rango` $: T \to \mathbb{N}$ y `ops` $: T \to \mathbb{N}$, que cuentan la altura de los árboles y la cantidad de operadores que ocurren en él. Por ejemplo

$$\texttt{rango}(((0 \otimes 1) \oplus (1 \oplus 0))) = 2 \qquad \texttt{ops}(((0 \otimes 1) \oplus (1 \oplus 0))) = 3$$

b. Pruebe las afirmaciones siguientes que sean correctas, y proporcione contraejemplos para las que no lo sean.

$$(\bar{\forall} t \in T)\texttt{rango}(t) \geq \texttt{ops}(t)$$
$$(\bar{\forall} t \in T)\texttt{rango}(t) > \texttt{ops}(t)$$
$$(\bar{\forall} t \in T)\texttt{rango}(t) \leq \texttt{ops}(t)$$

c. Defina recursivamente la función `eval` $: T \to \mathbb{N}$ que devuelve el natural representado por el árbol, interpretando $\oplus$ como la suma y $\otimes$ como el producto. Por ejemplo,

$$\texttt{eval}(((10100 \oplus 000100) \otimes 000100)) = 96$$

d. Considere un nuevo conjunto V de variables, y extendamos con ella el alfabeto Σ. Ahora agreguemos a la definición de T la siguiente cláusula adicional.

IV Si $v \in V$, entonces $v \in T$

Redefina la función `eval` $: T \times (V \to \mathbb{N}) \to \mathbb{N}$ que devuelve el natural representado por el árbol, donde agregamos un contexto o estado que proporciona información acerca de las variables. Por ejemplo, si $f(v_1) = 20$ y $f(v_2) = 4$, entonces $\texttt{eval}(((v_1 \oplus 000100) \otimes v_2), f) = 96$.

Ejercicio 21 (Otros esquemas)

Considere las siguientes listas de ecuaciones[3]

$$\begin{array}{ccc}
g(0) = 1 & h(0) = 1 & j(0) = 1 \\
g(1) = 0 & h(1) = 0 & j(2n+2) = 2 \times j(n) - 1 \\
g(n+2) = g(n) + \min\{g(n), g(n+1)\} & h(n+2) = h((n+2) \mod 2) & j(2n+1) = 2 \times j(n) - 1
\end{array}$$

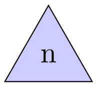
a. Determine si son formas correctas de definir funciones. Justifique.

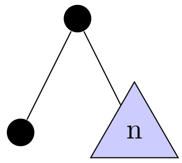
b. Pruebe que las primeras dos columnas definen una misma función.

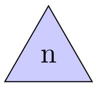
[3]Escribimos $n \mod m$ para indicar el resto de la división de n entre m.

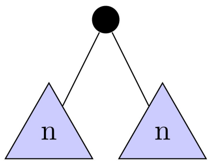
Ejercicio 22 (Árboles)

Considere las siguientes reglas de construcción de árboles binarios (sin información):

1. $\bullet \in \mathcal{L}$
2. Si 

$\in \mathcal{L}$, entonces 

$\in \mathcal{L}$
3. Si 

$\in \mathcal{L}$, entonces 

$\in \mathcal{L}$

Defino inductivamente las siguientes representaciones de naturales:

- el lenguaje N_1, definido inductivamente por las cláusulas 1 y 2.
- el lenguaje N_2, definido inductivamente por las cláusulas 1 y 3.
- el lenguaje N_3, definido inductivamente por las cláusulas 1 y 2 y 3.
- el lenguaje N_4, definido inductivamente por las cláusulas 2 y 3.

Cada árbol de los lenguajes anteriores codifica (o representa) al natural correspondiente a la altura del mismo; por ejemplo, la cláusula base de N_1 codifica el cero.

a. Indique si las siguientes afirmaciones son verdaderas o falsas:
   I. El lenguaje N_1 está incluído en el lenguaje N_3.
   II. El lenguaje N_2 está incluído en el lenguaje N_4.
   III. El lenguaje N_3 está incluído en el lenguaje N_1.
   IV. El lenguaje N_3 está incluído en el lenguaje $N_1 \cup N_2 \cup N_4$.

b. En caso que sea posible, construya una función $f : N_1 \to N_2$ siguiendo el ERP correspondiente que convierta las codificaciones de naturales en N_1 a sus correspondientes codificaciones en N_2.

c. En caso que sea posible, construya una función $f : N_2 \to N_3$ siguiendo el ERP correspondiente que convierta las codificaciones de naturales en N_2 a sus correspondientes codificaciones en N_3.

d. En caso que sea posible, construya una función $f : N_3 \to N_4$ siguiendo el ERP correspondiente que convierta las codificaciones de naturales en N_3 a sus correspondientes codificaciones en N_4.

e. En caso que sea posible, construya una función $f : N_3 \to N_2$ siguiendo el ERP correspondiente que convierta las codificaciones de naturales en N_3 a sus correspondientes codificaciones en N_2.

Observe que no todos los lenguajes definidos son libres, por lo que debe tener cuidado al usar el ERP.

Ejercicio 23 (Collatz)

Considere las siguientes ecuaciones

$$
\begin{aligned}
c(0) &= 1 \\
c(1) &= 1 \\
c(2 \times n) &= c(n) \\
c(2 \times n + 1) &= c(2 \times n \times k + k + 1)
\end{aligned}
$$

Llamamos c_i a las ecuaciones anteriores cuando reemplazamos k por i.

a. Demuestre que la función constante uno satisface las ecuaciones c_k, cualquiera sea k.
b. Demuestre que c_0 define la función constante uno.
c. Demuestre que c_1 define la función constante uno.
d. Demuestre que las ecuaciones c_2 no definen una función.
e. Escriba un programa en su lenguaje favorito que compute la función c_3.

Hoy en día, aún se desconoce si las ecuaciones c_3 definen la función constante uno. Esto es una variante menor de la conjetura de Collatz.