

# Guía 1 - Funciones primitivas recursivas y clases PRC

Solución de un alumno

Verano 2021

## Ejercicio 1

## Ejercicio 2

$$g_1(x) = x \dot{-} 1$$

$$g_1(0) = 0$$

$$g_1(t+1) = t = u_2^2(g(t), t)$$

$$g_3(x, y) = \max\{x, y\} = x \cdot (x > y) + y \cdot (x \leq y)$$

donde  $x \leq y$  es  $\alpha(x \dot{-} y)$  y  $x > y$  es  $\neg\alpha(x \dot{-} y)$

## Ejercicio 3

a) ( $\Rightarrow$ )

Qvq todas las funciones de la clase  $C_c$  se pueden escribir como  $f(\bar{x}) = k$  o  $f(\bar{x}) = x_i + k$  con un  $i$  fijo.

Demostremos por inducción estructural:

**CASO BASE:** Se ve que las iniciales cumplen.

**PASO INDUCTIVO:** Sea  $h$  una función obtenida por composición de funciones de  $C_c$ , entonces  $h$  tiene la pinta de:

$$h(\bar{x}) = f(g_1(\bar{x}), \dots, g_m(\bar{x})) \Rightarrow h(\bar{x}) = g_i(\bar{x}) + k \text{ ó } h(\bar{x}) = k$$

El segundo caso ya cumple. En el primer caso:

$$h(\bar{x}) = g_i(\bar{x}) + k h(\bar{x}) = x_j(\bar{x}) + k' + k = x_j(\bar{x}) + k'' \text{ ó } h(\bar{x}) = k' + k = k''$$

Cumpliendo las dos formas.

( $\Leftarrow$ )

$f(\bar{x}) = k$  está en  $C_c$  por ej1.

$f(\bar{x}) = x_i + k = u_i^n(\bar{x}) + k$  entonces está en  $C_c$  por composición.

b)  $g_2(x, y) = x \dot{-} y$  se puede demostrar con a).

## Ejercicio 4

$$\begin{aligned}\leq) f(x, y) &= \alpha(x \dot{-} y) \\ \geq) f(x, y) &= \alpha(y \dot{-} x) \\ <) f(x, y) &= \neg \alpha(y \dot{-} x) \\ >) f(x, y) &= \neg \alpha(x \dot{-} y) \\ =) f(x, y) &= (x \leq y) \cdot (y \leq x) \\ \neq) f(x, y) &= \neg[(x \leq y) \cdot (y \leq x)]\end{aligned}$$

siendo  $\neg(x) = \alpha(x)$

## Ejercicio 6

$$par(x) = \begin{cases} 1 & \text{si } x \text{ es par} \\ 0 & \text{si no} \end{cases}$$

$$par(0) = 1$$

$$par(x+1) = \neg par(x)$$

## Ejercicio 7

## Ejercicio 8

## Ejercicio 9

## Ejercicio 10

$$f(x) = \begin{cases} 0 & x = 0 \\ 1 & x = 1 \\ f(x-1) + f(x-2) & x > 1 \end{cases}$$

Observamos que cada elemento de la secuencia utiliza dos anteriores, entonces podemos crear una función similar pero que nos devuelva dos valores:

$$F(0) = \langle 0, 1 \rangle$$

$$F(n+1) = \langle r(F(n)), r(F(n)) + l(F(n)) \rangle$$

Entonces:

$$F(0) = \langle 0, 1 \rangle$$

$$F(1) = \langle 1, 1 \rangle$$

$$F(2) = \langle 1, 2 \rangle$$

$$F(3) = \langle 2, 3 \rangle$$

$$F(4) = \langle 3, 5 \rangle$$

$$F(5) = \langle 5, 8 \rangle$$

Y siempre vamos acumulando la suma de fibonacci a la derecha y a la izquierda nos guardamos un valor auxiliar (el anterior valor de la sucesión) para construir el siguiente.

Veamos que  $F$  es p.r.:

$<>, r(), l()$  y  $+$  son p.r., entonces  $F$  es p.r. por ser una recursiva primitiva de composiciones.

Entonces  $f(x)$  de fibonacci si la definimos por composición nos queda:

$$f(n) = r(F(n-1)) \cdot (n > 0)$$

## Ejercicio 11

$$h_1(\bar{x}, t) = \begin{cases} f_1(\bar{x}) & \text{si } t = 0 \\ g_1(h_1(\bar{x}, t-1), h_2(\bar{x}, t-1), \bar{x}, t) & \text{si no} \end{cases}$$

$$h_2(\bar{x}, t) = \begin{cases} f_2(\bar{x}) & \text{si } t = 0 \\ g_2(h_2(\bar{x}, t-1), h_1(\bar{x}, t-1), \bar{x}, t) & \text{si no} \end{cases}$$

Utilicemos la misma idea que el ejercicio de fibonacci, nos guardamos en la primer componente el valor de  $h_1$  y en la segunda el valor de  $h_2$ .

$$H(0, x) = \langle f_1(x), f_2(x) \rangle$$

$$H(n+1, x) = \langle g_1(l(H(n, x)), r(H(n, x)), \dots), g_2(r(H(n, x)), l(H(n, x)), \dots) \rangle$$

$H(x, y)$  pertenece a la clase  $C$  por ser recursiva primitiva con composición de funciones que estaban en  $C$ .

Entonces:

$$h_1(x, t) = l(H(t, x))$$

$$h_2(x, t) = r(H(t, x))$$

Y ambas están en  $C$ .

## Ejercicio 12

## Ejercicio 13

a) INYECTIVA: Dos secuencias diferentes van a tener diferente factorización de primos (si no termina en ceros) entonces van a tener diferente número natural asociado (porque la factorización de primos es única para cada número natural).

SOBREYECTIVA: Todo número natural mayor que cero tiene factorización de primos, entonces tiene una secuencia asociada.

b)

$$|\cdot| = \max_{x \leq} \{x : nprimo(x) | \cdot\}$$

$$\cdot[i] = \max \{x : nprimo(i)^x | \cdot\}$$

$$[x] = nprimo(1)^x$$

$$\cdot_1 \circ \cdot_2 = \cdot_1 \cdot \prod_{i=1}^m nprimo(i + |\cdot_1|)^{\cdot_2[i]}$$

$$sub(s, i, j) = \prod_{z=1}^j nprimo(z)^{s[z+i]}$$

c) Proponemos codificar una lista como:

$$[a_1, \dots, a_n] = \langle \prod_{i=1}^n nprimo(i)^{a_i}, n \rangle$$

Siendo la parte izquierda la codificación de primos anterior y la parte derecha la longitud de la lista, así pudiendo guardar listas que terminen con ceros. Veamos que los naturales y las secuencias forman una biyección.

INYECTIVA: Dos secuencias diferentes van a tener diferente factorización de primos o diferente longitud. Cuando alguno de ambos es diferente entonces la codificación de pares va a ser diferente (porque es inyectiva).

SOBREYECTIVA: como la codificación de pares es sobreyectiva, entonces existen  $a, b$  tal que  $c = \langle a, b \rangle$  entonces toda codificación tiene su propia secuencia asociada.

Ahora no importa si la secuencia termina con ceros, va a tener su número propio.

$$|\cdot| = r(\cdot)$$

$$\cdot[i] = l(\cdot)[i]$$

$$[\cdot] = \langle l(\cdot), 1 \rangle$$

$$\cdot_1 \circ \cdot_2 = \langle l(\cdot_1) \circ l(\cdot_2), r(\cdot_1) + r(\cdot_2) \rangle$$

$$sub(x, i, j) = \langle sub(l(x)), j - i + 1 \rangle$$

## Ejercicio 14

a) Misma idea que fibonacci pero ahora nos tenemos que guardar todos los valores anteriores.

$$H(x, 0) = [f([], x)]$$

$$H(x, t) = H(x, t-1) \circ [f(H(x, t-1), x)]$$

Entonces:

$$h(x, t) = H(x, t)[|H(x, t)|]$$

b) Casi igual al a).