Boletín 2: Funciones recursivas primitivas

Computabilidad y Complejidad (3CO21)
Mario Campos Mocholí - macammoc@inf.upv.es

1

Sea una función recursiva primitiva $g : N^{k+1} \longrightarrow N$. Se define la función de N^k a N

$$\tau x.g(x,\ n_1,\ \ldots,\ n_k) = \begin{cases} m,\ si\ (\exists\ i \in \{0,\ \ldots,\ x\})\ (g(i,\ n_1, \ldots,\ n_k)\ =\ m \\ \quad \wedge\ (\forall j:\ 0 \leq j < i)\ (g(j,\ n_1,\ \ldots,\ n_k)\ =\ 0)) \\ 0,\ en\ otro\ caso \end{cases}$$

Demuestre que la función $\tau x.g(x, n_1, \ldots, n_k)$ es recursiva primitiva.

Se define la función $\tau x.g$ como:

$$\tau x.g(0,\ n_1, \ldots,\ n_k) = desigual(g(0,\ n_1, \ldots,\ n_k), 0)) \cdot g(0,\ n_1, \ldots,\ n_k)$$
$$\tau x.g(s(x),\ n_1, \ldots,\ n_k) = desigual(g(s(x),\ n_1, \ldots,\ n_k),\ 0))$$
$$\cdot\ igual(\sum_{i=0}^{x} g(i,\ n_1, \ldots,\ n_k),\ 0) \cdot g(s(x),\ n_1, \ldots,\ n_k)$$
$$+\ cosg(desigual(g(s(x),\ n_1, \ldots,\ n_k),\ 0) \cdot igual(\sum_{i=0}^{x} g(i,\ n_1, \ldots,\ n_k),\ 0)) \cdot \tau x.g(x,\ n_1, \ldots,\ n_k)$$

En el caso de que x sea 0 comprueba que el resultado de $g(0, ...)$ sea m devolviendo este como valor o 0 caso contrario. Como caso general, comprueba que el resultado de $g(x, ...)$ sea m y que para todo i menor que x y mayor o igual que 0, el resultado de aplicar $g(i, ...)$ sea 0, devolviendo m. En caso contrario el $cosg(...)$ del segundo elemento de la suma vale 1 y se realizaría la llamada a $\tau(x-1).g$. La función acaba cuando $x = 0$ ó cuando se ha encontrado una i que cumple las condiciones.

Las funciones empleadas se pueden encontrar en:
$suma(n,\ m)$ - pág. 21 "Tema 3: Funciones Mu-Recursivas"
$prod(n,\ m)$ - pág. 24 "Tema 3: Funciones Mu-Recursivas"
$cosg(n,\ m$ - pág. 27 "Tema 3: Funciones Mu-Recursivas"
$igual(n,\ m)$ - pág. 31 "Tema 3: Funciones Mu-Recursivas"
sumatorio - pág. 35 "Tema 3: Funciones Mu-Recursivas"

2

Sean f, g y h funciones recursivas primitivas definidas de N a N y sea k un número natural predeterminado. Se define la función p como sigue

$$p(n) = \begin{cases} f(g(h(n-2)+1)+1) & si \;\; n > 1 \\ k & si \; n \leq 1 \end{cases}$$

¿Es p una función recursiva primitiva?

Definimos la función p como:

$$p(0) = k$$
$$p(1) = k$$
$$p(n) = f(aux3(n))$$

Siendo las funciones auxiliares utilizadas:

$$aux3(n) = s(aux2(n))$$
$$aux2(n) = g(aux1(n))$$
$$aux1(n) = s(aux0(n))$$
$$aux0(n) = h(dif(n,\ 2))$$

Si n es menor o igual a 1 se devuelve directamente k. En caso contrario se aplica la función sustitución, definiendo diversas funciones auxiliares para mantener el esquema de recursividad, y devolviendo el valor de la combinación de estas.

Las funciones empleadas se pueden encontrar en:
$sucesor(n)$ - pág. 6 ”Tema 3: Funciones Mu-Recursivas”
composición - pág. 10 ”Tema 3: Funciones Mu-Recursivas”
$dif(n,\ m)$ - pág. 23 ”Tema 3: Funciones Mu-Recursivas”

3

Sea la función $primo : N \longrightarrow N$, definida de modo que

$$primo(n) = \begin{cases} 1, & si\ n\ es\ primo \\ 0, & en\ otro\ caso \end{cases}$$

Demuestre que $primo$ es una función recursiva primitiva.

Definimos la función $primo$ como:

$$primo(0) = 0$$
$$primo(1) = 0$$
$$primo(n) = igual(\sum_{i=0}^{n}(cosg(resto(n,\ i)),\ 3)$$

Si n es 0 ó 1 directamente se devuelve 0. En cualquier otro caso se comprueba que el sumatorio del signo contrario del resto de n entre los i menores o iguales que n es igual a 3. Un número es primo si este sumatorio da 3 ya que se suma el caso de $i = 0$, el caso de $i = 1$ y el caso de $i =$ n. Si suma más de 3 es que se ha encontrado un i divisor de n y por tanto este no es primo.

Las funciones empleadas se pueden encontrar en:
$cosg(n,\ m$ - pág. 27 "Tema 3: Funciones Mu-Recursivas"
$igual(n,\ m)$ - pág. 31 "Tema 3: Funciones Mu-Recursivas"
sumatorio - pág. 35 "Tema 3: Funciones Mu-Recursivas"
$resto(n,\ m)$ - pág. 1 "Boletín de ejercicios de autoevaluación 2"

4

Sean f y g dos funciones primitivo recursivas y se define la función h como sigue

$$h(n) = \begin{cases} f(n)^{g(n+1)}\ si\ f(n) > g(n) \\ g(n)^{f(n+1)}\ si\ f(n) \leq g(n) \end{cases}$$

¿Es h una función recursiva primitiva?

Demostramos que h es una función recursiva primitiva mediante la siguiente definición:

$$\begin{aligned} h(n) = {} & mayor(f(n),\ g(n))\ \cdot\ exp(g(s(n)),\ f(n)) \\ & + meig(f(n),\ g(n))\ \cdot\ exp(f(s(n),\ g(n))) \end{aligned}$$

Si $f(n)$ es mayor que $g(n)$, el resultado de la función *mayor* valdrá 1 y se multiplicara por el resultado de $f(n)$ elevado a $g(n\ +\ 1)$. En caso contrario, *mayor* valdrá 0 y *meig* valdrá 1 por lo que el resultado será $g(n)$ elevado a $f(n\ +\ 1)$. Como f y g son funciones recursivas primitivas y todas las empleadas lo son también, h será una función recursiva primitiva.

Las funciones empleadas se pueden encontrar en:
$suma(n,\ m)$ - pág. 21 "Tema 3: Funciones Mu-Recursivas"
$prod(n,\ m)$ - pág. 24 "Tema 3: Funciones Mu-Recursivas"
$mayor(n,\ m)$ - pág. 30 "Tema 3: Funciones Mu-Recursivas"
$exp(g,\ m)$ - pág. 37 "Tema 3: Funciones Mu-Recursivas"
$meig(n,\ m)$ - pág. 1 "Boletín de ejercicios de autoevaluación 2"

5

Se define la función $f: N \times N \longrightarrow N$ de modo que $f(n,m)$ es igual al producto de los divisores de n que son pares y mayores que m y menores que n, si los hay, o 1 en otro caso.
Demuestre que f es una función recursiva primitiva.

Para demostrar que f es una función recursiva primitiva la definimos como:

$$\prod_{i=0}^{n}(cosg(resto(n,\ i)) \cdot par(i) \cdot menor(i,\ n) \cdot mayor(i,\ m) \cdot i$$
$$+\ cosg(cosg(resto(n,\ i)) \cdot par(i) \cdot menor(i,\ n) \cdot mayor(i,\ m)))$$

Se realiza el productorio de los números i tales que cumplen la condición de ser divisores de n (el resto es 0), ser pares, menores que n y mayores que m. Si al comprobar un i este no cumple alguna condición, la primera parte de la suma valdrá 0 y la segunda 1, y lo contrario para cuando cumpla todas. De esta forma se asegura no multiplicar por 0 y devolver 1 si no hay ningún elemento que cumpla las condiciones.

Las funciones empleadas se pueden encontrar en:
$suma(n,\ m)$ - pág. 21 "Tema 3: Funciones Mu-Recursivas"
$prod(n,\ m)$ - pág. 24 "Tema 3: Funciones Mu-Recursivas"
$cosg(n,\ m$ - pág. 27 "Tema 3: Funciones Mu-Recursivas"
$menor(n,\ m)$ - pág. 29 "Tema 3: Funciones Mu-Recursivas"
$mayor(n,\ m)$ - pág. 30 "Tema 3: Funciones Mu-Recursivas"
$productorio$ - pág. 35 "Tema 3: Funciones Mu-Recursivas"
$resto(n,\ m)$ - pág. 1 "Boletín de ejercicios de autoevaluación 2"