UNIVERSIDAD DE OVIEDO
ESCUELA POLITÉCNICA DE
INGENIERÍA de GIJÓN

ALGORITMIA
2ª Evaluación Teórica: Diseño Recursivo
10 de noviembre de 2021

1) DISEÑO RECURSIVO [7.5 puntos] Diseñar un algoritmo recursivo que resuelva el siguiente problema: "Dada una matriz de enteros A[1..n][1..m] siendo n≥1 y m≥1, diseñar una función recursiva que retorne cierto si se cumple que para cada fila el número de ceros que contiene es menor o igual que el número de ceros que contiene la fila inmediatamente siguiente, y falso en caso contrario. Responder exclusivamente a las siguientes cuestiones, con claridad y concisión:

a) Especificación formal de la función: precondición, postcondición y perfil de la función (15%)
b) Describir el principio de inducción utilizado (5%)
c) Realizar una inmersión no final, escribir la especificación formal de la función inmersora e indicar cómo se ha de efectuar la primera llamada a la función (10%)
d) Sucesor, o sucesores, de un problema dado (5%)
e) Establecer las condiciones de caso/s trivial/es y caso/s no trivial/es (5%)
f) Explicar razonadamente la solución propuesta para el/los caso/s trivial/es, es decir, *triv*($\bar{x}$) (15%)
g) Explicar razonadamente la solución propuesta para el/los caso/s no trivial/es, es decir, explicar la función de combinación *c*($\bar{x}$, $\overline{y'}$) (25%)
h) Escribir la función inmersora en pseudocódigo, incluyendo función/es auxiliar/es si se precisan (20%)

$A[1..3][1..4] = \begin{bmatrix} 1 & 2 & 4 & 8 \\ 0 & 6 & 0 & 24 \\ 0 & 10 & 0 & 0 \end{bmatrix}$ Solución: TRUE $\qquad A[1..3][1..4] = \begin{bmatrix} 1 & 0 & 4 & 8 \\ 3 & 6 & 12 & 24 \\ 4 & 10 & 0 & 0 \end{bmatrix}$ Solución: FALSE

SOLUCIÓN.-

a) Especificación formal de la función: precondición, postcondición y perfil de la función

$Q \equiv \{\, n, m \geq 1 \,\}$

Función Evaluacion (A[1..n][1..m]:matriz de enteros) retorna (b : booleano)

$R \equiv \{\, b = (\forall i)((\mathrm{N}j)(A[i-1][j] = 0 : 1 \leq j \leq m) \leq (\mathrm{N}\, j)(A[i][j] = 0 : 1 \leq j \leq m) : 2 \leq i \leq n)\}$

b) Describir el principio de inducción utilizado (5%)

Inducción Noetheriana: A cada matriz le asociaremos un natural n que corresponde al número de filas de la matriz. Seguidamente aplicaremos inducción simple sobre dicho natural.

c) Realizar una inmersión no final, escribir la especificación formal de la función inmersora e indicar cómo se ha de efectuar la primera llamada a la función (10%)

Sustituimos n por k en R:

$Q' \equiv \{\, n, m \geq 1 \ y \ 1 \leq k \leq n \,\}$

Función iEvaluacion (A[1..n][1..m]: matriz de enteros, k: entero) retorna (b: booleano)

$R' \equiv \left\{\, b = (\forall i)\left((\mathrm{N}j)(A[i-1][j] = 0 : 1 \leq j \leq m) \leq (\mathrm{N}\, j)(A[i][j] = 0 : 1 \leq j \leq m) : 2 \leq i \leq k\right)\right\}$

La llamada inicial a la función iEvaluacion se realiza con k = n, es decir,

Evaluacion(A) = iEvaluacion(A, n)

d) Sucesor, o sucesores, de un problema dado (5%)

Dado el problema iEvaluacion(A, k), para pasar al siguiente problema “eliminamos” una fila. Así, el subproblema sucesor será iEvaluacion(A, k - 1).

e) Establecer las condiciones de caso/s trivial/es y caso/s no trivial/es (5%)

Bt ≡ k = 1 la sección de la matriz tiene una única fila.

Bnt ≡ k > 1 la sección de la matriz tiene dos o más filas.

f) Explicar razonadamente la solución propuesta para el/los caso/s trivial/es, es decir, *triv*($\bar{x}$) (15%)

Cuando la matriz está formada por una única fila, es decir, al sustituir en R´ k por 1, la R´ resultante sería la siguiente:

$$b = (\forall i)\big((\mathrm{N}j)(A[i-1][j] = 0 : 1 \le j \le m) \le (\mathrm{N}\, j)(A[i][j] = 0 : 1 \le j \le m) : 2 \le i \le 1\big)$$

Como se puede observar el cuantificador ∀ tiene un rango vacío, y por lo tanto el resultado será VERDADERO (elemento neutro de la operación que representa).

g) Explicar razonadamente la solución propuesta para el/los caso/s no trivial/es, es decir, explicar la función de combinación *c*($\bar{x}$, $\overline{y'}$) (25%)

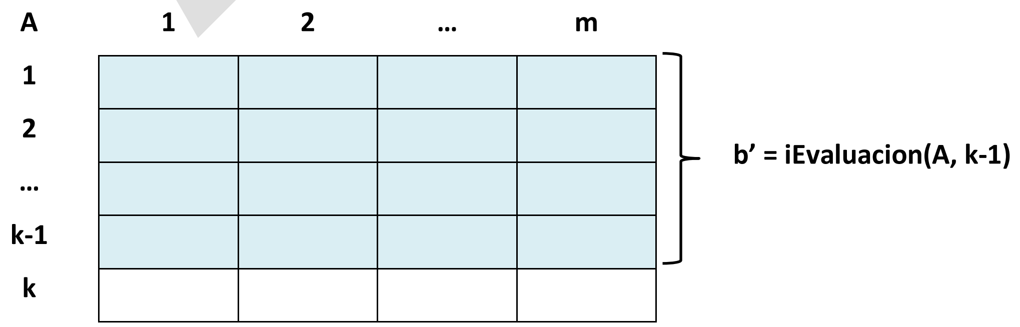
La solución del problema iEvaluacion(A, k) consiste en determinar si se cumple la propiedad para la submatriz A[1..k][1..m], es decir,

$$b = (\forall i)\big((\mathrm{N}j)(A[i-1][j] = 0 : 1 \le j \le m) \le (\mathrm{N}\, j)(A[i][j] = 0 : 1 \le j \le m) : 2 \le i \le k\big)$$

Hipótesis de recurrencia: Supongamos conocida la solución del subproblema iEvaluacion(A, k-1). Llamaremos a dicho valor, b’. Por lo tanto, suponemos conocida la solución para la submatriz A[1..k-1][1..m], es decir,

$$b' = (\forall i)\big((\mathrm{N}j)(A[i-1][j] = 0 : 1 \le j \le m) \le (\mathrm{N}\, j)(A[i][j] = 0 : 1 \le j \le m) : 2 \le i \le k-1\big)$$

Gráficamente sería



Tesis:

En consecuencia, para obtener la solución del problema iEvaluacion(A, k) a partir de la solución del subproblema iEvaluacion(A,k-1) habría que comprobar además que el número de ceros que contiene la fila k-1 es menor o igual que el número de ceros que contiene la fila k, es decir,

$$(\mathrm{N}j)(A[k-1][j] = 0 : 1 \leq j \leq m) \leq (\mathrm{N}\, j)(A[k][j] = 0 : 1 \leq j \leq m)$$

Por lo tanto, iEvaluacion(A, k) = iEvaluacion(A, k-1) y

$$((\mathrm{N}j)(A[k-1][j] = 0 : 1 \leq j \leq m) \leq (\mathrm{N}\, j)(A[k][j] = 0 : 1 \leq j \leq m))$$

h) Escribir la función inmersora en pseudocódigo, incluyendo función/es auxiliar/es si se precisan (20%)

```
Función iEvaluacion ( A[1..n][1..m]:matriz de enteros; k: entero ) retorna ( b : booleano )
   si k = 1 entonces retorna VERDADERO
            sino retorna iEvaluacion ( A, k-1 ) AND AUX ( A, k )
   fsi
función
```

donde la función AUX sería la siguiente:

```
Función AUX ( A[1..n][1..m]: matriz de entero; k: entero) retorna ( b : booleano )
var j, ceros_fila_anterior, ceros_fila_actual : entero fvar;
      ceros_fila_anterior = 0
      ceros_fila_actual = 0

      para j = 1 hasta m hacer
           si ( A[ k-1][ j ] = 0 )
              entonces ceros_fila_anterior = ceros_fila_anterior + 1;
           fsi
           si (A[ k][ j ] = 0)
              entonces ceros_fila_actual = ceros_fila_actual + 1;
           fsi
      fpara
    si ( ceros_fila_anterior <= ceros_fila_actual ) retorna VERDADERO  sino retorna FALSO  fsi
ffuncion
```

2.- Responder a las siguientes cuestiones:

a) **[1 punto]** Demostrar el paso 3 de verificación formal para la función que se muestra a continuación. La función iCapicua, vista en clase expositiva, comprueba si un vector V[1..n] con n≥1 es capicúa o no, esto es, si es el mismo leído de izquierda a derecha que de derecha a izquierda. La llamada inicial a la función sería iCapicua(V,1,n):

$Q' \equiv \{1 \leq i \leq j \leq n\}$

```
Función iCapicua ( V[1..n] : vector de enteros; i, j : entero ) retorna (s : booleano)
 caso
      i = j    → VERDADERO
      i = j-1 → ( V[ i ] = V[ j ] )
      i < j-1 → iCapicua( V, i+1, j-1 )  AND  ( V[ i ] = V[ j ] )
 fcaso
ffuncion
```

$R' \equiv \{s = (\forall k)(V[\,k\,] = V[\,j - k + i\,]: i \leq k \leq j)\}$

Solución:

3. $Q(\overline{x}) \wedge B_t(\overline{x}) \Rightarrow R(\overline{x}, triv(\overline{x}))$

Caso base: i = j

$1 \leq i \leq j \leq n \wedge i = j \Rightarrow$ VERDADERO = $(\forall k)$ (V[k] = V[j – k + i]: $i \leq k \leq j$)

Partimos de la postcondición, esto es,

$$s = (\forall k) (V[\,k\,] = V[\,j - k + i\,]: i \leq k \leq j)$$

sustituimos s por VERDADERO y j por i y tendremos

$$\text{VERDADERO} = (\forall k) (V[\,k\,] = V[\,i - k + i\,]: i \leq k \leq i)$$

$$\text{VERDADERO} = (V[\,i\,] = V[\,i - i + i\,])$$

$$\text{VERDADERO} = (V[\,i\,] = V[\,i\,])$$

$$\text{VERDADERO} = \text{VERDADERO}$$

Sí, se cumple

Caso base: i = j-1

$(1 \le i \le j \le n) \wedge (i = j\text{-}1) \Rightarrow (V[i] = V[j]) = (\forall k)(V[k] = V[j - k + i]: i \le k \le j)$

Partimos de la postcondición, esto es,

$$s = (\forall k)(V[k] = V[j - k + i]: i \le k \le j)$$

sustituimos s por (V[i] = V[j]) e i por j-1 y tendremos

$$(V[j\text{-}1] = V[j]) = (\forall k)(V[k] = V[j - k + (j - 1)]: j - 1 \le k \le j)$$

$$(V[j\text{-}1] = V[j]) = (V[j - 1] = V[j - (j - 1) + (j - 1)]) \wedge (V[j] = V[j - j + (j - 1)])$$

$$(V[j\text{-}1] = V[j]) = (V[j - 1] = V[j]) \wedge (V[j] = V[j - 1])$$

$$(V[j\text{-}1] = V[j]) = (V[j - 1] = V[j])$$

Sí, se cumple

b) **[0.5 puntos]** ¿Cómo definirías la función t (función limitadora) si tuvieras que demostrar los apartados 5 y 6 de verificación formal para la función iCapicua?

$$t(V, i, j) \stackrel{\text{def}}{=} j - i + 1$$

c) **[0.5 puntos]** Si en lugar de n≥1 hubiera sido n≥0, ¿las condiciones que determinan los casos triviales y el caso no trivial cambiarían? En caso afirmativo, indicar cómo serían bajo este supuesto (n≥0) y en caso negativo, explicar por qué no hay que realizar cambios.

Sí, Bt y Bnt cambiarían para contemplar el rango vacío. Las nuevas expresiones booleanas serían:
$Bt \equiv (i > j)$ y $Bnt \equiv (i \le j)$

d) **[0.5 puntos]** Seguimos en el mismo supuesto del apartado c), esto es, n≥0, ¿la precondición de la función iCapicua cambiaría? En caso afirmativo, indicar cómo sería bajo este supuesto (n≥0) y en caso negativo, explicar por qué no hay que realizar cambios.

Sí, Q´ cambiaría para contemplar el rango vacío. La nueva Q´ sería:

$$Q' \equiv \{1 \le i \le j + 1 \le n + 1\}$$