

## METODOLOGÍA DE LA PROGRAMACIÓN

Grado en Ingeniería Informática de Gestión y Sistemas de Información

Escuela de Ingeniería de Bilbao (UPV/EHU)

Departamento de Lenguajes y Sistemas Informáticos

Curso: 1º – Curso académico: 2018-19

14 de febrero de 2019

Examen parcial – Tema 2: Documentación – Grupo 01 – 2 puntos

### EJERCICIO 1 – (1,100 puntos)

- a) (0,075 puntos) Definir el predicado **mayor\_igual**(G(1..r), H(1..r), x) que exprese que, posición a posición, si el valor de G(1..r) es un 1, el valor de H(1..r) es mayor o igual que x.

**Ejemplo:**

Los vectores G(1..8) y H(1..8) de este ejemplo cumplen **mayor\_igual**(G(1..8), H(1..8), 20)

G(1..8)	5	1	1	12	4	1	0	1
	1	2	3	4	5	6	7	8

H(1..8)	10	88	23	12	50	20	3	40
	1	2	3	4	5	6	7	8

- b) (0,025 puntos) Definir el predicado **bits**(I(1..r)) que exprese que el vector I(1..r) solo contiene ceros y unos.

**Ejemplo:** El vector I(1..8) de este ejemplo cumple **bits**(I(1..8))

I(1..8)	0	0	1	0	1	1	1	1
	1	2	3	4	5	6	7	8

- c) (0,100 puntos) Definir el predicado

**division**(D(1..r), (d<sub>1</sub>, d<sub>2</sub>, ..., d<sub>r</sub>), E(1..r), (e<sub>1</sub>, e<sub>2</sub>, ..., e<sub>r</sub>), F(1..r), (f<sub>1</sub>, f<sub>2</sub>, ..., f<sub>r</sub>), pos)

que exprese lo siguiente:

- pos es mayor o igual que 0 y menor o igual que r.
- En las posiciones k comprendidas entre 1 y pos (las posiciones 1 y pos incluidas):
  - Si f<sub>k</sub> es 1, entonces el valor de D(k) es d<sub>k</sub> / e<sub>k</sub>, el valor de E(k) es (((d<sub>k</sub> / e<sub>k</sub>) + 1) \* e<sub>k</sub>) – d<sub>k</sub> y el valor de F(k) es f<sub>k</sub>. Por tanto, en D(k) se tiene el resultado de la división entera entre d<sub>k</sub> y e<sub>k</sub> y en E(k) se tiene la diferencia (positiva) entre d<sub>k</sub> y el primer múltiplo de e<sub>k</sub> que es mayor que d<sub>k</sub>.
  - Si f<sub>k</sub> no es 1, entonces el valor de D(k) es d<sub>k</sub>, el valor de E(k) es e<sub>k</sub> y el valor de F(k) es f<sub>k</sub>.

**Ejemplo:**

Los vectores D(1..8), (5, 19, 10, 20, 10, 30, 38, 40), E(1..8), (8, 4, 15, 6, 15, 20, 32, 43), F(1..8) y (30, 1, -5, 1, 1, 15, -4) de este ejemplo cumplen lo siguiente:

**division**(D(1..8), (5, 19, 10, 20, 10, 30, 38, 40), E(1..8), (8, 4, 15, 6, 15, 20, 32, 43), F(1..8), (30, 1, -5, 1, 1, 15, -4), 5)

siendo D(1..8), E(1..8) y F(1..8) los siguientes vectores:

	19 / 4		20 / 6		10 / 15			
D(1..8)	5	4	10	3	0	30	38	17
	1	2	3	4	5	6	7	8

	α		β		δ			
E(1..8)	8	1	15	4	5	20	32	-6
	1	2	3	4	5	6	7	8

F(1..8)	30	1	-5	1	1	1	15	-4
	1	2	3	4	5	6	7	8

donde  $\alpha = (((19/4) + 1) * 4) - 19$ ,  $\beta = (((20/6) + 1) * 6) - 20$  y  $\delta = (((10/15) + 1) * 15) - 10$

Para esos mismos vectores, no se cumple:

**division**(D(1..8), (5, 19, 10, 20, 10, 30, 38, 40), E(1..8), (8, 4, 15, 6, 15, 20, 32, 43), F(1..8),  
(30, 1, -5, 1, 1, 1, 15, -4), 6)

porque en la posición 6 los vectores D, E y F deberían tener los valores 1 (30/20), 10 (((30/20) + 1) \* 20) - 30 y 1 respectivamente, en vez de los valores 30, 20 y 1.

d) (0,900 puntos) **Documentar**, dando las fórmulas que se cumplen en los puntos indicados y utilizando los predicados definidos en los apartados anteriores, el siguiente programa que:

- Recibe como datos de entrada tres vectores de enteros A(1..n), B(1..n) y C(1..n) donde  $n \geq 1$  y donde, C(1..n) solo contiene ceros y unos, y posición a posición, siempre que en C(1..n) se tenga un 1, en A(1..n) se tiene un valor mayor o igual que 0 y en B(1..n) se tiene un valor mayor o igual que 1.
- Como resultado, para cada posición k comprendida entre 1 y n, si el valor inicial de C(k) es 1, A(k) contendrá el resultado de la división entera de los valores iniciales de A(k) y B(k); por su parte, B(k) contendrá la diferencia (positiva) entre el valor inicial de A(k) y el primer múltiplo del valor inicial de B(k) que es mayor que el valor inicial de A(k); en cuanto a C(k), conserva su valor inicial. Si el valor inicial de C(k) no es 1, los tres vectores conservan su valor inicial.

(1) {Precondición}	0,060	Puntuación que corresponde a cada fórmula ←
i := 1;	0,005	
(2) {Aserción intermedia}	0,200	
<b>while</b> (3) {Invariante} i ≤ n <b>loop</b>	0,010	
(4) {Aserción intermedia}	0,010	/ es la división entera. Ejemplos: 9 / 4 = 2 12 / 9 = 1 8 / 4 = 2
<b>if</b> C(i) = 1 <b>then</b>	0,010	
(5) {Aserción intermedia}	0,050	
aux := A(i) / B(i);	0,150	
(6) {Aserción intermedia}	0,150	
B(i) := ((aux + 1) * B(i)) - A(i);	0,050	
(7) {Aserción intermedia}	0,100	
A(i) := aux;	0,090	
(8) {Aserción intermedia}	0,025	
<b>end if</b> ;		
(9) {Aserción intermedia}		
i := i + 1;		
(10) {Aserción intermedia}		
<b>end loop</b> ;		
(11) {Postcondición}		
(12) {Expresión cota E}		

## EJERCICIO 2 – (0,900 puntos)

- a) (0,020) Definir el predicado **multiplo(x, y)** que exprese que **x** es un múltiplo de **y**, es decir, el resto de dividir **x** por **y** es cero.

### Ejemplos:

multiplo(8, 4) es True porque el resto de dividir el número 8 por 4 es 0. En cambio, multiplo(9, 4) es False porque el resto de dividir el número 9 por 4 no es 0. De la misma forma, multiplo(4, 8) es False porque el resto de dividir el número 4 por 8 no es 0.

- b) (0,880 puntos) **Documentar** dando las fórmulas que se cumplen en los puntos indicados y utilizando el predicado definido en el apartado anterior, el siguiente programa que:

- Recibe como datos de entrada cuatro números enteros p, q, v y w donde p es mayor o igual que 1, q es mayor o igual que p + 1, v es mayor o igual que q + 1 y w es mayor o igual v.
- Como resultado, devuelve en la variable *mc* el número de múltiplos comunes de p y q en el intervalo [v..w].

(1) {Precondición}	0,010	Puntuación que corresponde a cada fórmula ←
h := v - 1;		
(2) {Aserción intermedia}	0,005	
mc := 0;		
(3) {Aserción intermedia}	0,005	
<b>while</b> (4) {Invariante} $h \leq w - 1$ <b>loop</b>	0,210	
(5) {Aserción intermedia}	0,030	
aux := (((h + 1) mod p) = 0) <b>and</b> (((h + 1) mod q) = 0);		
(6) {Aserción intermedia}	0,040	
<b>if</b> aux = True <b>then</b>		
(7) {Aserción intermedia}	0,040	mod es el resto de la división entera. Ejemplos: 9 mod 4 = 1 12 mod 9 = 3 8 mod 4 = 0
mc := mc + 1;		
(8) {Aserción intermedia}	0,180	
<b>end if;</b>		
(9) {Aserción intermedia}	0,050	
h := h + 1;		
(10) {Aserción intermedia}	0,180	
<b>end loop;</b>		
(11) {Postcondición}	0,100	
(12) {Expresión cota E}	0,030	