

## Estructuras de Datos y Algoritmos

Grados en Ingeniería Informática, de Computadores y del Software

Examen Parcial, 11 de Febrero de 2013.

1. **Diseño iterativo (4 puntos)** Especificar, diseñar, verificar y calcular el coste de un algoritmo que, dado un vector  $v$  de enteros que puede ser vacío ( $n \geq 0$ ) y que tan solo puede contener los valores 0 y 1, devuelva un booleano que indique si el vector tiene la forma:

$$1 | 1 | \dots | 1 | 1 | 0 | 0 | \dots | 0 | 0$$

Las dos zonas de ceros y unos pueden tener cualquier longitud, incluida la nula.

*Sugerencia (no penaliza si no se sigue):* la verificación puede resultar más fácil si se elige un invariante de la forma:

$$1 | \dots | 1 | ? ? ? ? ? | 0 | \dots | 0$$

2. **Diseño recursivo (4 puntos)** Se define el *histograma* de un vector  $v$  de enteros como otro vector  $w$  que contiene en la posición  $i$  la suma  $v[0] + \dots + v[i]$ . Así, por ejemplo, el vector  $\{1, 2, 4, 3, -2\}$  tendría como histograma  $\{1, 3, 7, 10, 8\}$ .

Se pide especificar, diseñar, demostrar la corrección y calcular el coste de un algoritmo recursivo, lo más eficiente posible, que dado un vector de enteros que puede ser vacío devuelva su histograma.

3. **Diseño TADs (2 puntos)** Diseñar un TAD *Polinomio* que permita manejar polinomios con coeficientes enteros en una indeterminada. Ejemplos de polinomios serían  $3x^4 + 1$ ,  $-2x^{99} + 5x^2$ , ó  $x^{507} + x^{200} + x$ . El tipo *Polinomio* deberá permitir, además de la construcción de polinomios, su suma, resta y multiplicación (que devolverán nuevos polinomios); su evaluación para una  $x$  concreta; y la posibilidad de verificar si un polinomio está vacío o es igual a otro dado.

Se pide:

- Elegir un *tipo representante* para el TAD eficiente en tiempo y espacio, suponiendo que los polinomios están formados por  $k$  o menos monomios (elementos de la forma  $c \cdot x^n$ ), donde tanto  $c$  como  $n$  pueden llegar a ser muy grandes (aunque quepan en un `int` estándar). La constante  $k$  debe formar parte de la implementación de tu TAD.

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

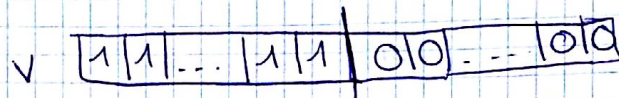
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

y describas correctamente.

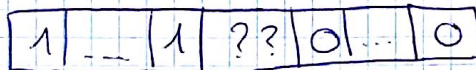


Feb 2013

① Algoritmo que detecte



Invariante



Especificación:

$\{ n \geq 0 \wedge n = \text{long}(v) \wedge \forall w: 0 \leq w < n (v[w] = 0) \vee v[w] = 1) \}$

fun unoseros (int v[], int n) return bool b;

$\{ b = [ \exists i: 0 \leq i < n: [ (\forall t: 0 \leq t < i: v[t] = 1) \wedge (\forall s: i \leq s < n: v[s] = 0) ] \wedge$

$\wedge [ (\# w: 0 \leq w < n \cdot v[w] = 1) = (\# r: 0 \leq r < n \cdot v[r] = 0) ] \}$

Diseño:

```
bool unoseros (int v[], int n) {
    bool b;
    int i;
    b = true;
    i = 0;
    while (i < n-1) {
        b = b && (v[i] == 1) && (v[n-i-1] == 0);
        i = i+1;
    }
}
```

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

---

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70



# Verificación

## Invariante

$$I \equiv (0 \leq 2i \leq n) \wedge (\forall w: 0 \leq w < i: (v[w] = 1 \wedge v[n-w-1] = 0))$$

$$B \equiv 2i < n$$

$$= b$$

**I1**

$$\{P\} A_0 \{I\}$$

$$\{P\} b = t, i = 0 \{I\}$$

$$\text{pmd } (b = t, i = 0, I) \Leftrightarrow I \Big|_{i=0}^{\text{true}} \Big|_b \Leftrightarrow$$

$$\Leftrightarrow (0 \leq 0 \leq n) \wedge (\forall w: \underbrace{0 \leq w < 0}_{\text{false}}: \dots) = \text{true} \Leftrightarrow (0 \leq n) \leftarrow P$$

**I2**

$$\{I \wedge B\} b = b \ \&\& \ v[i] = 1 \ \&\& \ v[n-1-i] = 0; i = i + 1$$

$$\text{pmd } (\_ ) \Leftrightarrow I \Big|_{i}^{i+1} \Big|_b^{b \wedge \_}$$

$$\Leftrightarrow (0 \leq 2(i+1) \leq n) \wedge ((\forall w: 0 \leq w < i+1: v[w] = 1 \wedge v[n-w-1] = 0) \wedge v[i] = 1 \wedge v[n-1-i] = 0)$$

$$\Leftrightarrow (0 \leq 2(i+1) \leq n) \wedge (b = (\forall w: 0 \leq w < i+1: v[w] = 1 \wedge v[n-w-1] = 0))$$

$$I \wedge B \equiv (0 \leq 2 \leq n) \wedge (b = (\forall w: 0 \leq w < i: v[w] = 1 \wedge v[n-w-1] = 0)) \wedge (2i < n)$$



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

---

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70



I3

$$I \wedge \neg B \Rightarrow Q$$

$$(0 \leq z_i \leq n+1) \wedge (b = \dots) \wedge (z_i \geq n)$$

$$\Rightarrow (n \leq z_i \leq n+1) \wedge (b = \dots)$$

$$\Rightarrow [(z_i = n) \vee (z_i = n+1)] \wedge (b = \dots)$$

$$\Rightarrow [(z_i = n) \wedge (b = \dots)] \vee [(z_i = n+1) \wedge (b = \dots)]$$

$$\Rightarrow [(z_i = n) \wedge (b = \forall w : 0 \leq w < \frac{n}{2} : v[w] = 1 \wedge v[n-w-1] = 0)]$$

$$\vee [(z_i = n+1) \wedge (b = \forall w : 0 \leq w < \frac{n+1}{2} : v[w] = 1 \wedge v[n-w-1] = 0)]$$

$$\Rightarrow (z_i = n) \wedge (b = \forall w : 0 \leq w < \frac{n}{2} : v[w] = 1 \wedge v[n-w-1] = 0)$$

Q

### Función de cota

$$C(v, n, b, i) = n - z_i$$

C1

$$I \wedge B \Rightarrow C > 0$$

$$(0 \leq z_i \leq n+1) \wedge (\dots) \wedge (z_i < n)$$

$$\Rightarrow 0 < n - z_i$$

$$\Rightarrow 0 \leq n - z_i$$

C2

$$\langle I \wedge B \wedge C = T \rangle \wedge \langle C < T \rangle$$

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

---

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# Complejidad

```
bool ...
```

```
  b = true;
```

```
  i = 0;
```

```
  while (i < n - i)
```

```
    b = b & v[i] == 1 && v[n - i - 1] == 0;
```

```
    i = i + 1;
```

```
  }
```

```
  return b;
```

```
}
```

$$T(n) = 11 \frac{n}{2} ; \text{ si } n \text{ par}$$

$$T(n) = 11 \frac{n+1}{2} ; \text{ si } n \text{ impar}$$

$\left. \begin{array}{l} \\ \\ \end{array} \right\} \in \Theta(n)$

n par:  $\left(\frac{n}{2}\right)$  vueltas

$n=2$

i	n-i
0	2
1	1

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

---

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70



## ② Histograma

Dado un vector  $v[N]$ , su histograma es otro vector  $h[N]$  tq  $h[i] = v[0] + \dots + v[i]$

### Especificación

$$P \equiv \{0 \leq N \wedge \text{long}(v) = N\}$$

func histograma (int  $v[]$ , int  $N$ ) return (int  $h[]$ )

$$Q = \{\forall i: 0 \leq i < N: h[i] = \sum_{w: 0 \leq w \leq i} v[w]\}$$

### Generalizamos

$$P \equiv \{0 \leq N \wedge \text{long}(v) = N \wedge \text{long}(h) = N \wedge 0 \leq k < N \wedge \text{sumParc} = \sum_{i=0}^{k-1} v[i]\}$$

proc histograma (int  $v[]$ , int  $N$ , int  $k$ , int  $h[]$ , int sumP)

$$Q = \{\forall i: 0 \leq i < N: h[i] = \sum_{w: 0 \leq w \leq i} v[w]\}$$

### Diseño

```
void histograma (int  $v[]$ , int  $N$ , int  $k$ , int  $h[]$ , int sumP) {
```

```
// if (k == N) nada (nos hemos salido del vector)
```

```
if (k < N) {  
    sumP = sumP +  $v[k]$ ;
```

```
     $h[k]$  = sumP;
```

```
    histograma ( $v$ ,  $N$ ,  $k+1$ ,  $h$ , sumP)
```

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

Cartagena99



• Paso 1: Cálculo para un caso partic.  $n \in \mathbb{Z}$

$$T(n) = 6 + T(n-1)$$

$$= 6 + 6 + T(n-2) = 2 \cdot 6 + T(n-2)$$

$$= 2 \cdot 6 + 6 + T(n-3) = 3 \cdot 6 + T(n-3)$$

= ... n pasos ...

$$= 6n + T(0)$$

$$= 6n$$

• Paso 2: Conjetura

$$T(n) = 6n$$

- CASO BASE ( $n=0$ )  $T(0) = 0 = 6 \cdot 0 \checkmark$

lo supongo para  $n' < n$

- CASO GENERAL

Hip  $\forall n' < n \quad T(n') = 6n'$

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

---

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70