

Apuntes-Tema-2.pdf



Juandf03



Análisis y diseño de algoritmos



2º Grado en Ingeniería Informática



**Escuela Técnica Superior de Ingeniería Informática
Universidad de Málaga**



Escuela de
organización
industrial

La mejor escuela de negocios en
energía, sostenibilidad y medio
ambiente de España.

Más información
www.eoi.es

Formamos
talento para un futuro
Sostenible



100% Empleabilidad



Modalidad: Presencial u online



**Programa de Becas,
Bonificaciones y Descuentos**

Tema 2

"Especificación"

1. Especificación. Implementación

Especificación → que hace el algoritmo
bajo qué condiciones } lenguaje lógico matemático

Implementación → secuencia de instrucciones

¡Ej!

$\{ a \geq 0, b > 0 \}$

$\{ a = b \cdot q + z, z < b \}$

⇒

```
int q = 0;
int z = a;
while ( z >= b ) {
    z = z - b;
    q++;
}
```

2. Especificación pre / post

→ Aserio : expresión que se evalúa a cierto o a falso (predicado)

→ Estado : valores definidos de las variables (o)

• Aserio precondición $pre \equiv \{ \}$ → caracteriza los estados iniciales válidos

• Aserio postcondición $post \equiv \{ \}$ → relación válida entre datos de entrada y salida

¡Ej!

$Pre \equiv \{ a \geq 0, b > 0 \}$

$Post \equiv \{ a = b \cdot q + z, z < b \}$



3. Predicados lógicos ... expresión algebraica + operador relacional

$<, \leq, >, \geq, =$

Varias operadores \Rightarrow operador lógico ($\neg, \vee, \wedge, \dots$)

Quantificadores $\Rightarrow \forall, \exists, \Sigma, \dots$

• Variables ligadas (vs) Variables libres

ej.

$$P \equiv (x + y + z > 0)$$

$$Q \equiv (\exists y : y \in \mathbb{N} ; P) \longrightarrow \begin{cases} \text{libres}(Q) = \{x, z\} \\ \text{ligadas}(Q) = \{y\} \end{cases}$$

\downarrow
tal que

4. Semántica

$[[P]]_{\sigma} \longrightarrow$ resultado de evaluar un predicado P en un estado σ

- SATISFACIBLE $\exists \sigma ; \sigma \in \Sigma \Rightarrow \sigma \models P$
- VÁLIDO $\forall \sigma ; \sigma \in \Sigma \Rightarrow \sigma \models P (\models P)$
- CONTRADICCIÓN $\forall \sigma ; \sigma \in \Sigma \Rightarrow \sigma \not\models P$

* Q es una consecuencia lógica de $P \rightarrow P \models Q$

$$\forall \sigma ; \sigma \in \Sigma ; (\sigma \models P \Rightarrow \sigma \models Q)$$

5. Especificación con predicados

El conjunto de estados definido por P :

$$\text{estados}(P) = \{ \sigma \in \Sigma ; [[P]]_{\sigma} = \text{cierto} \}$$



**Nosotros te dejamos la cara perfecta,
pon tu la cara dura y dale caña al juego.**

No hace falta que te la juegues con un suspenso,
ya te traemos el juego nosotros para que salgas ganando.

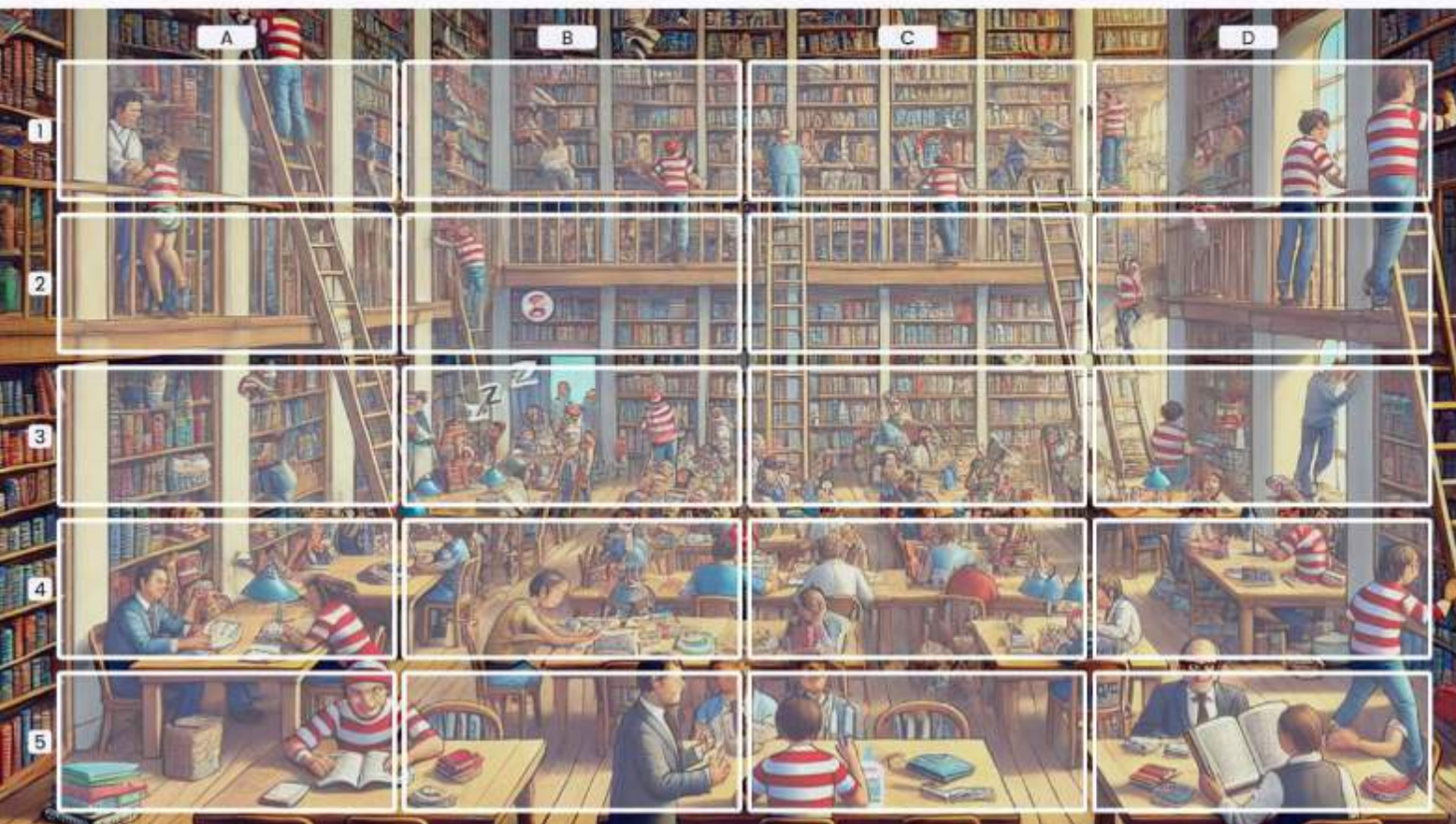
REGLAS

1. Encuentra el producto oculto en el anuncio dentro de tu apunte.
2. Escanea el QR para acceder al juego en Wuolah.
3. Introduce la coordenada donde se esconde el producto.
4. Gana tu recompensa 🏆

Fácil 5



¡Juega ahora!



Análisis y diseño de algoritmos



Comparte estos flyers en tu clase y consigue más dinero y recompensas



Banco de apuntes de la

- 1** Imprime esta hoja
- 2** Recorta por la mitad
- 3** Coloca en un lugar visible para que tus compis puedan escanar y acceder a apuntes

- 4** Llévate dinero por cada descarga de los documentos descargados a través de tu QR



→ Predicados equivalentes ($P \equiv Q$) si la satisfacen los mismos estados

6. Especificación de problemas

- Reforzar y debilitar un predicado

conjunción (\wedge)

disyunción (\vee)

- Convenios y notación

$$\{a..b\} = \begin{cases} \emptyset & \text{si } a > b \\ \{a\} \cup \{a+1..b\} & \text{si } a \leq b \end{cases}$$

$N(P(x))$

$x \in \{a..b\}$

→ n° de veces que ocurre algo

¡ig!

$$N(a[i] \neq 0) = \square$$

$i \in \{0..a.length-1\}$

Tema 3

"¿Dónde y Vencerás?"

1. Introducción

Se divide el problema en instancias más pequeñas, las resolvemos y combinamos las soluciones.

↓
¡RECURSIVIDAD!

$$T(n) = a T\left(\frac{n}{b}\right) + f(n)$$

① $\frac{n}{b}$ tamaño subproblema

② a n° subproblemas

③ $f(n)$ complejidad combinar subproblemas
los soluciones