



UNIVERSITAT  
POLITÈCNICA  
DE VALÈNCIA

# Recogida de paquetes en un edificio<sup>1</sup>

Alfons Juan

*DSIC*

Departamento de Sistemas  
Informáticos y Computación

---

<sup>1</sup>Para una correcta visualización, se requiere Acrobat Reader v. 7.0 o superior

# Índice

<b>1</b>	<b>Recogida de paquetes en un edificio</b>	<b>1</b>
<b>2</b>	<b>Esquema diseño SBR</b>	<b>3</b>
<b>3</b>	<b>El edificio pequeño: Anchura</b>	<b>4</b>
<b>4</b>	<b>El edificio pequeño: Profundidad</b>	<b>5</b>

# 1. Recogida de paquetes en un edificio

Edificio 11x4 con:

- Un robot (# tiros)
- Escaleras (E)
- Cajas (C)
- “Enemigos” (X)

4			C				X				
3	E			C			E				E
2		E		X			X		E	C	
1	2		E				E				
	1	2	3	4	5	6	7	8	9	10	11

**Espacio estados:** posición y tiros robot, info cajas y enemigos

**Info estática:** dimensiones edificio; posición escaleras y huecos

**Estado inicial:** figura de arriba

**Acciones:** dcha, izqda, arriba, abajo, recoger y disparar

**Objetivo:** recoger todas las cajas

Las reglas han de servir para diferentes dimensiones del edificio, así como diferente número y posición de escaleras, huecos, cajas y enemigos. Nada más consideramos un único robot.

# Recogida (cont.): anchura y profundidad

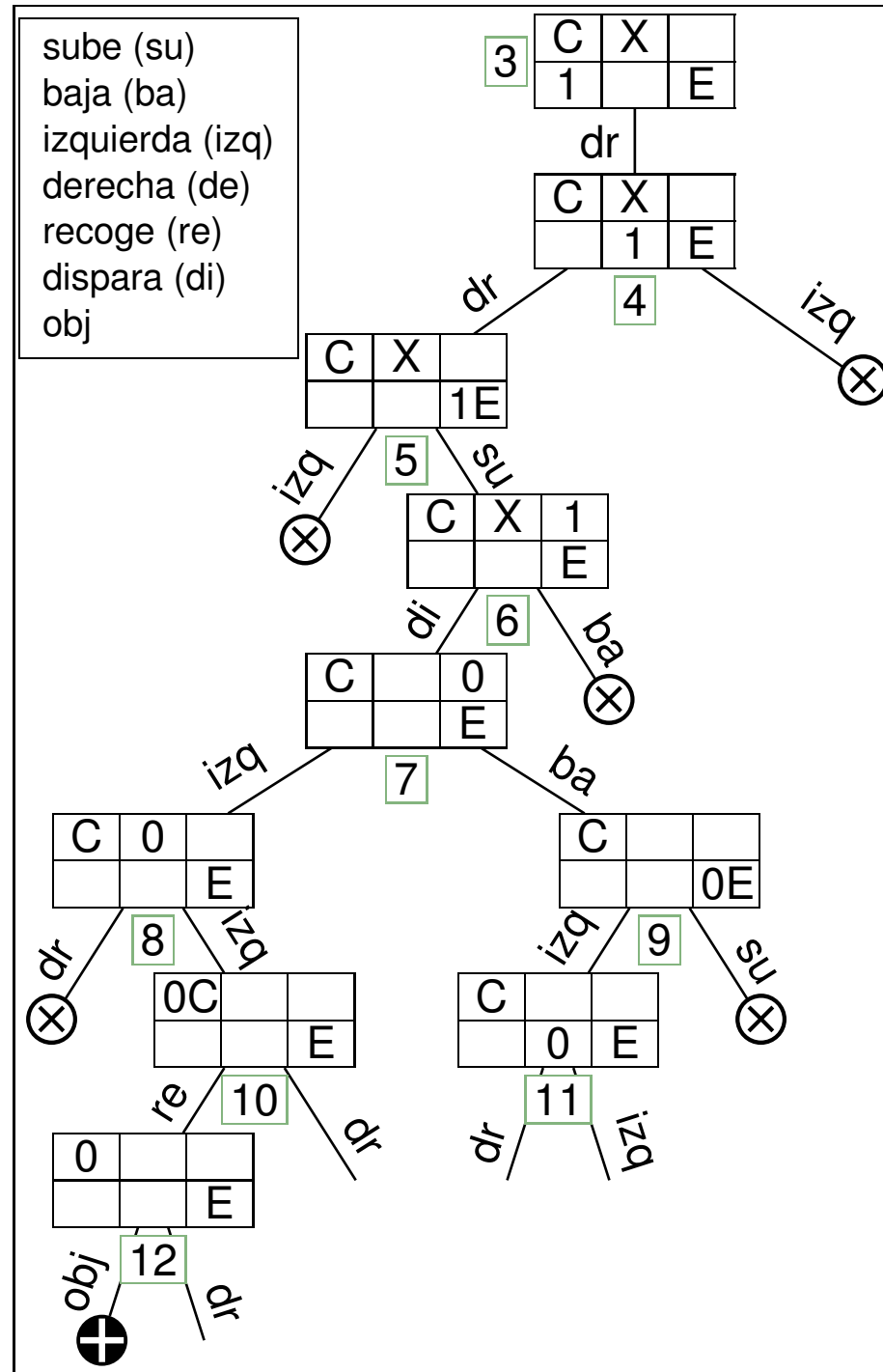
## 2. Esquema diseño SBR

x-skel.clp

```
; Base de hechos:
(deffacts bf ...))

; Reglas:
(defrule sube
  "robot en escalera" => "robot sube")
(defrule baja
  "robot arriba de escalera" => "robot baja")
(defrule izquierda
  "robot puede ir a la izquierda" => "robot a la izquierda")
(defrule derecha
  "robot puede ir a la derecha" => "robot a la derecha")
(defrule recoge
  "robot en caja" => "robot recoge caja")
(defrule dispara
  "robot puede disparar" => "robot dispara")
(defrule obj
  (declare (salience 1))
  "no quedan cajas" => "fin")
```

### 3. El edificio pequeño: Anchura



## 4. El edificio pequeño: Profundidad

