

# ESPECIFICACIÓN USANDO TADs

## EJEMPLOS DE COMPORTAMIENTO AUTOMÁTICO

Departamento de Computación,  
Facultad de Ciencias Exactas y Naturales,  
Universidad de Buenos Aires

Algoritmos y Estructuras de Datos II

## ASCENSOR AUTOMÁTICO

Se quiere especificar el comportamiento de un ascensor que lleva personas entre dos pisos. La capacidad máxima del ascensor es de 3 personas. El ascensor se pone en funcionamiento cuando ingresan 3 personas, sin necesidad de apretar ningún botón. Cuando el ascensor se pone en movimiento, se desplaza del piso en el que está hacia el otro. En cuanto llega al piso de destino, las personas que están en el interior del ascensor lo desocupan inmediatamente.

En cualquier momento pueden llegar personas a cualquiera de los dos pisos. En el caso que el ascensor no esté, las personas forman una fila y esperan a que el ascensor las venga a buscar.

## REQUERIMIENTOS

Se quiere saber:

- En qué piso está el ascensor
- La cantidad de personas esperando en cada uno de los dos pisos

Tenemos que definir:

- 1 Los observadores y la igualdad observacional.
- 2 Los generadores.
- 3 Las otras operaciones (si hay).
- 4 Axiomatizar todo.

# ¿QUÉ QUEREMOS MODELAR?

- 1 ¿Qué nos están pidiendo sobre las personas? ¿Nos interesa identificarlas?
- 2 En consecuencia, ¿hace falta modelar la fila de personas?
- 3 ¿Nos interesa diferenciar a las personas que están esperando dentro y fuera del ascensor?

Entonces los observadores elegidos son:

<pre>personasEn : ascensor × piso  → nat piso?      : ascensor  → piso</pre>
------------------------------------------------------------------------------

# GENERADORES PROPUESTA 1

crearAscensor :  $\longrightarrow$  ascensor

llegaPersona : ascensor  $\times$  piso  $\longrightarrow$  ascensor

cambiarPiso : ascensor  $a \times$  piso  $p \longrightarrow$  ascensor

$\{p = \text{elOtroPiso}(\text{piso?}(a)) \wedge \text{personasEn}(a, \text{elOtroPiso}(p)) \geq 3\}$

Encuentre el error!

- Si  $p$  solamente puede tomar el valor del *piso en el que no está el ascensor*, ¿por qué es un parámetro?
- El enunciado dice "*El ascensor se pone en funcionamiento cuando ingresan 3 personas*"
  - ¿Entonces por qué dejamos que haya más de 3 personas en el piso del ascensor?

Encuentre el error

crearAscensor :  $\longrightarrow$  ascensor

llegaPersona : ascensor  $a \times$  piso  $p \longrightarrow$  ascensor  
 $\{p \neq \text{piso?}(a) \vee \text{personasEn}(a, p) < 3\}$

cambiarPiso : ascensor  $a \times$  piso  $p \longrightarrow$  ascensor  
 $\{p = \text{elOtroPiso}(\text{piso?}(a)) \wedge \text{personasEn}(a, \text{elOtroPiso}(p)) = 3\}$

- La clase no se llamaba *comportamiento automático*?
- Entonces el cambio de piso debería ser automático
- Es decir, no debería existir una instancia del TAD en el que haya 3 personas en el piso del ascensor y éste no se mueva
- Más específicamente, el cambio de piso es algo que tiene que reflejarse en los axiomas y no en los generadores

Encuentre el error

Generadores

$\text{crearAscensor} : \text{piso } p \times \text{nat } \textit{personasPB} \times \text{nat } \textit{personas1} \longrightarrow \text{ascensor}$   
 $\{(p=\text{PB} \wedge \textit{personasPB} < 3) \vee (p=\text{1ero} \wedge \textit{personas1} < 3)\}$

Otras operaciones

$\text{llegaPersona} : \text{ascensor } a \times \text{piso } p \longrightarrow \text{ascensor}$

- *Es formalmente correcto*
- *Pero sería mucho problema axiomatizar la operación  $\text{llegaPersona}$*

Los generadores elegidos son:

`crearAscensor` :  $\longrightarrow$  ascensor

`llegaPersona` : ascensor  $\times$  piso  $\longrightarrow$  ascensor



A axiomatizar todo

- 1 ¿Cómo repercutió el comportamiento automático en la elección de los *observadores*?
  - **No lo hizo.** Para distinguir lo que diferenciaba a los distintos ascensores no importó si el comportamiento era automático o no
- 2 ¿Cómo repercutió el comportamiento automático en los *generadores*?
  - Tuvimos que quitar el generador que cambiaba el piso del ascensor.
- 3 ¿Cómo repercutió el comportamiento automático en los *axiomas*?
  - **En casi todo.** El comportamiento queda plenamente descrito en los axiomas.

## EL ASCENSOR 2.0

Muchas veces la gente se cansa de esperar el ascensor, dado que mientras haya menos de 3 personas en el piso donde está el ascensor, éste no va a moverse. Esto implica que en el otro piso pueden acumularse muchísimas personas esperando.

Para esto se decidió incorporar un botón en cada piso para llamar al ascensor. Si el botón es presionado en el piso donde el ascensor se encuentra, nada sucede. Por el contrario, si se presiona en el piso donde el ascensor no está, el ascensor se mueve hacia el piso destino llevando con él a todas las personas que su capacidad le permita y que hayan estado esperando en el piso de origen.

Obviamente, si el botón es presionado es que hay al menos una persona en el piso donde se lo presionó.

- 1 Si los requerimientos son los mismos, ¿tenemos que cambiar los observadores? ¿Y la igualdad observacional?
- 2 ¿Qué pasa con los axiomas? Hay que pensar qué cambia en la versión 2.0 con respecto al comportamiento anterior

## EL ASCENSOR 3.0 (TAREA)

Otro problema que se está teniendo es que mientras no se alcanza la capacidad máxima y nadie del otro piso llama al ascensor, éste no se mueve. Esto implica que la gente dentro del ascensor se tiene que quedar esperando innecesariamente. Para esto, se instaló un botón dentro del ascensor que cuando es presionado el ascensor cambia de piso, llevándose a la gente que estaba esperando con él. Claramente, para que este botón pueda ser presionado, tiene que haber alguien dentro del ascensor.