



DEPARTAMENTO DE INFORMÁTICA
UNIVERSIDAD CARLOS III DE MADRID

Ingeniería en Informática

Inteligencia Artificial

Prueba de autoevaluación

Normas generales del examen

- La solución del examen se entregará en una sola hoja doble.
- La hoja de enunciado también deberá entregarse junto a la hoja de solución.
- Recordad incluir vuestro nombre, apellidos, y el grupo en ambas hojas, la del enunciado, y la de la solución.
- El tiempo para realizar el examen es de **30 minutos**
- No se responderá a ninguna pregunta sobre el examen
- Si se sale del aula, no se podrá volver a entrar durante el examen
- No se puede presentar el examen escrito a lápiz
- No se puede utilizar ningún tipo de documentación ni herramienta de cómputo automático

Apellidos Nombre(s) Grupo: ...

Problema 1. (1 punto)

Una empresa de tecnología industrial desea automatizar el proceso de decisión empleado para dirigir el movimiento de un ascensor en grandes edificios. Para ello, ha decidido emplear técnicas de **planificación automática** de tareas.

En los problemas que pretenden resolver, se considera la presencia de varias personas distribuidas por un mismo edificio que, en un momento dado, solicitan la atención del ascensor. Por lo tanto, se asume que, de cada persona, se conoce el piso en el que se encuentra, así como el piso al que desea ir.

Se pide:

- ① (0,25 puntos) Empleando la **lógica de predicados**, indicar qué predicados sirven para describir cada estado.
- ② (0,25 puntos) Pon un ejemplo de un **estado inicial** y un **estado final** con tres personas.
- ③ (0,5 puntos) Representar los operadores en lenguaje STRIPS.

Solución

- ① Existen varias formas de intentar resolver este problema, pero la más eficiente consiste en determinar qué personas ya han llegado a sus destinos. Estos hechos podrían almacenarse en los predicados `served (persona)` y `not-served (persona)`. Obviamente, para conseguir dirigir un planificador para que resuelva estos problemas, es preciso conocer la posición actual de cada persona (predicado `origin (persona, planta)`) y a qué planta desea llegar (predicado `destin (persona, planta)`), así como distinguir si se encuentra dentro de un ascensor (predicado `boarded (persona)`) o no —predicado `not-boarded (persona)`.

Por último, para poder determinar el movimiento del ascensor, bastará con conocer su posición (`lift-at (planta)`), así como la relación de precedencia entre las plantas, para lo que basta el predicado `above (planta-1, planta-2)`.

- ② En la definición de un estado inicial, imagínese que **Mónica**, **Gema** y **Fernando** desean reunirse todos en la planta número 3. Mientras que **Mónica** y **Gema** están en la planta 1, en el estado inicial dispondremos a **Fernando** en la planta 5 (de modo que nadie está dentro del ascensor):

```
origin (Mónica, 1)    origin (Gema, 1)    origin (Fernando, 5)
destin (Mónica, 3)   destin (Gema, 3)   destin (Fernando, 3)
not-boarded (Mónica) not-boarded (Gema) not-boarded (Fernando)
```

y el ascensor está, por ejemplo, en la planta 4:

```
lift-at (4)
```

y la relación de precedencia entre las plantas, en un edificio con 7 plantas por ejemplo, se establece con:

```
above (1,0)          above (2,1)          above (3,2)
above (4,3)          above (5,4)          above (6,5)
above (7,6)
```

como ninguno de ellos está en su planta destino, debe añadirse:

```
not-served (Mónica)  not-served (Gema)   not-served (Fernando)
```

Por último, el estado final consiste, simplemente, en solicitar que todas las personas lleguen a sus destinos:

```
served (Mónica)      served (Gema)        served (Fernando)
```

- ③ Básicamente, hay dos tipos diferentes de acciones que debe gobernar el planificador: el movimiento del ascensor (ya sea haciéndolo subir, `up (planta-1, planta-2)`, o bajar con el operador `down (planta-1, planta-2)`) y el acceso de las personas al ascensor —ya sea entrando en él con el operador `board (planta, persona)` o saliendo con el operador `depart (planta, persona)`. A continuación se detallan los operadores en el lenguaje STRIPS.

Primero, el ascensor puede dirigirse con los operadores:

`up (planta-1, planta-2):`

```
Precondiciones: lift-at (planta-1)           (verificar la posición original del ascensor)
                  above (planta-2, planta-1) (verificar que la planta-2 es accesible hacia arriba)
Añadidos:        lift-at (planta-2)         (el ascensor pasa a la planta superior)
Borrados:        lift-at (planta-1)         (y deja de estar en la inferior)
```

y:

down (planta-1, planta-2):

Precondiciones:	lift-at (planta-1)	<i>(verificar la posición original del ascensor)</i>
	above (planta-1, planta-2)	<i>(verificar que la planta-2 es accesible hacia abajo)</i>
Añadidos:	lift-at (planta-2)	<i>(el ascensor pasa a la planta superior)</i>
Borrados:	lift-at (planta-1)	<i>(y deja de estar en la inferior)</i>

Nótese que los dos operadores añaden y borran exactamente los mismos literales, sin embargo ha sido preciso distinguirlos para poder considerar diferentes precondiciones (precisamente la relación de precedencia entre las plantas). Esto ocurre típicamente en la modelización de operadores con STRIPS.

Por otra parte, la modelización del uso que las personas hacen del ascensor se puede automatizar con los operadores:

board (planta, persona):

Precondiciones:	lift-at (planta)	<i>(verificar la posición original del ascensor)</i>
	origin (persona, planta)	<i>(y de la persona)</i>
Añadidos:	boarded (persona)	<i>(ahora el ascensor tiene alguien más)</i>
Borrados:	No hay	

y:

depart (planta, persona):

Precondiciones:	lift-at (planta)	<i>(verificar la posición original del ascensor)</i>
	boarded (persona)	<i>(y que la persona está en el ascensor)</i>
	destin (persona, planta)	<i>(y que desea bajar aquí)</i>
Añadidos:	served (persona)	<i>(la persona ha llegado a su destino)</i>
Borrados:	boarded (persona)	<i>(y deja de estar en el ascensor)</i>

Nótese que en la solución del problema se ha asumido que el ascensor tiene capacidad infinita¹ y que siempre hay, como se advertía en el enunciado, un único ascensor —de modo que basta con decir si alguien está o no en el ascensor, sin necesidad de indicar cuál.

¹Para poder resolver este problema, debe extenderse el paradigma de planificación al caso de *Planificación Numérica*, cuya presentación y discusión está fuera del propósito de este curso.