Airbus Planning Problem
Planificación Automática
Trabajo Práctico

Antonio Quirós
Planificación Automática
Departamento de Informática
Escuela Politécnica Superior

Tomar asiento en un avión:

Caminar hasta tu fila, tomar el asiento de la letra que te corresponde...

# Fácil.
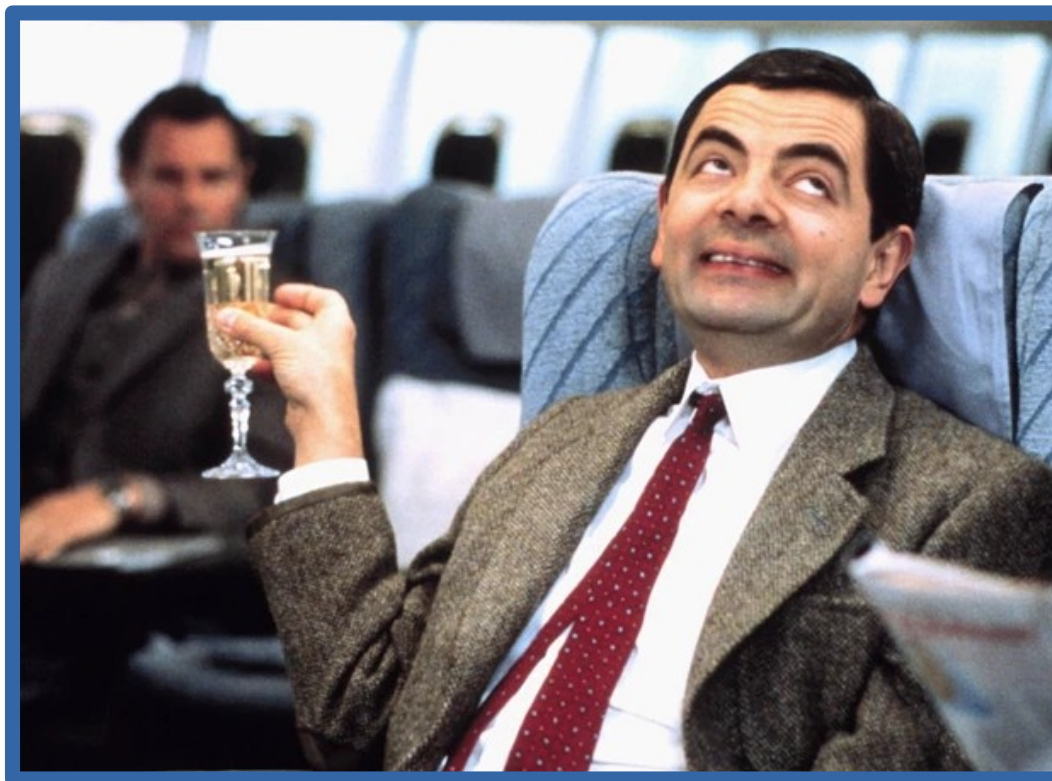
Pero...

No siempre es tán cómodo como parece.

Restricciones

1.- Para tomar asiento en un puesto, éste debe estar desocupado.

2.- Para tomar asiento en (o levantarse de) un puesto de ventana (A o F), los asientos del medio y del pasillo también deben estar desocupados.

3.- Para tomar asiento en (o levantarse de) un puesto del medio (B o E), el asiento del pasillo debe estar desocupado.

4.- Dos personas no pueden estar en el pasillo en la misma fila al mismo tiempo.

Distribución de los asientos
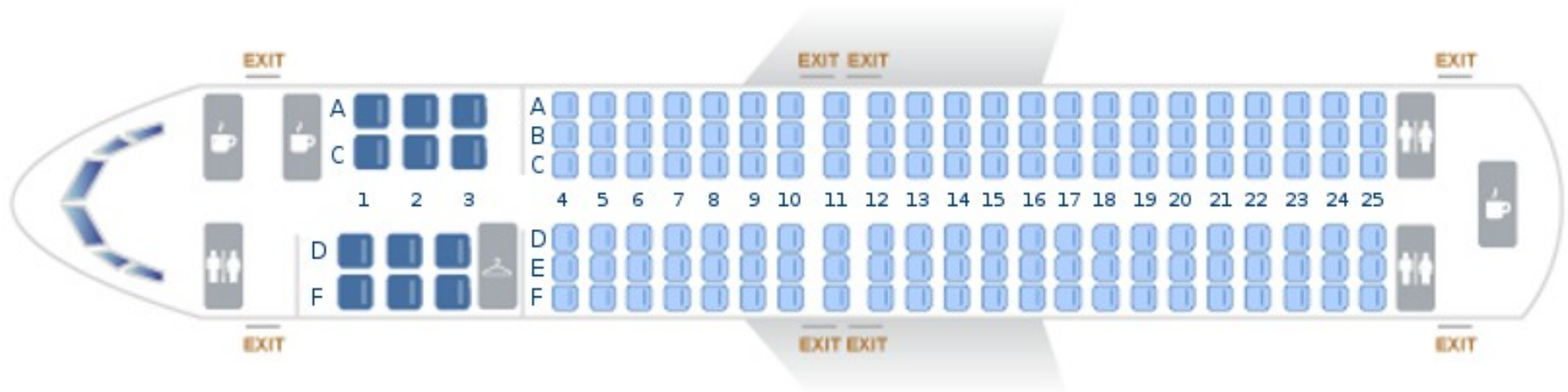
Subir pasajeros a un avión como un problema de planificación automática

Tipos:

row

passenger

Constantes:

A B C D E F

Predicados:

(**occupied** ?row – row ?seat – object)

(**passenger-seated** ?row – row ?seat – object ?passenger – passenger)

(**boarded** ?passenger – passenger)

(**at-hallway-row** ?row – row ?passenger – passenger)

(**hallway-blocked** ?row – row)

Subir pasajeros a un avión como un problema de planificación automática

Acciones:

```
sit_in_A, sit_in_B, sit_in_C, sit_in_D, sit_in_E, sit_in_F
stand_up_from_A, stand_up_from_B, stand_up_from_C, stand_up_from_D,
 stand_up_from_E, stand_up_from_F
walk-to-row
move-to-other-row
```

Subir pasajeros a un avión como un problema de planificación automática

```
(:action sit_in_A
    :parameters (?row – row ?passenger – passenger)
    :precondition (and
        (not (occupied ?row A))
        (not (occupied ?row B))
        (not (occupied ?row C))
        (at-hallway-row ?row ?passenger)
        )
    :effect (and
        (occupied ?row A)
        (passenger-seated ?row A ?passenger)
        (not (at-hallway-row ?row ?passenger))
        (not (hallway-blocked ?row))
        )
    )
```

Subir pasajeros a un avión como un problema de planificación automática

```
(:action sit_in_B
    :parameters (?row – row ?passenger – passenger)
    :precondition (and
        (not (occupied ?row B))
        (not (occupied ?row C))
        (at-hallway-row ?row ?passenger)
        )
    :effect (and
        (occupied ?row B)
        (passenger-seated ?row B ?passenger)
        (not (at-hallway-row ?row ?passenger))
        (not (hallway-blocked ?row))
        )
    )
```

Subir pasajeros a un avión como un problema de planificación automática

```
(:action stand_up_from_A
    :parameters (?row – row ?passenger – passenger)
    :precondition (and
        (occupied ?row A)
        (passenger-seated ?row A ?passenger)
        (not (occupied ?row B))
        (not (occupied ?row C))
        (not (hallway-blocked ?row))
        )
    :effect (and
        (not (occupied ?row A))
        (not (passenger-seated ?row A ?passenger))
        (hallway-blocked ?row)
        (at-hallway-row ?row ?passenger)
        )
    )
```

Subir pasajeros a un avión como un problema de planificación automática

```
(:action walk-to-row
    :parameters (?row – row ?passenger – passenger)
    :precondition (and
        (not (hallway-blocked ?row))
        (not (boarded ?passenger))
        )
    :effect (and (boarded ?passenger)
                 (hallway-blocked ?row)
                 (at-hallway-row ?row ?passenger)
                 )
)
```

Subir pasajeros a un avión como un problema de planificación automática

```
(:action move-to-other-row
    :parameters (?from - row ?to - row ?passenger - passenger)
    :precondition (and
        (not (hallway-blocked ?to))
        (boarded ?passenger)
        (at-hallway-row ?from ?passenger)
        )
    :effect (and (not (at-hallway-row ?from ?passenger))
            (not (hallway-blocked ?from))
            (hallway-blocked ?to)
            (at-hallway-row ?to ?passenger)
        )
    )
```

Subir pasajeros a un avión como un problema de planificación automática

Problema 1:
6 filas, 6 pasajeros


```
(define (problem airbus-seats-p01-6f-6p)
(:domain airbus-seats)
(:objects
    row1 row2 row3 row4 row5 row6 - row
    pass1 pass2 pass3 pass4 pass5 pass6 -
passenger)
(:init )
(:goal
    (and
    (occupied row2 A)
    (occupied row2 B)
    (occupied row2 C)
    (occupied row2 D)
    (occupied row2 E)
    (occupied row2 F)
    )
))
```

Subir pasajeros a un avión como un problema de planificación automática

Problema 2:
6 filas, 36 pasajeros
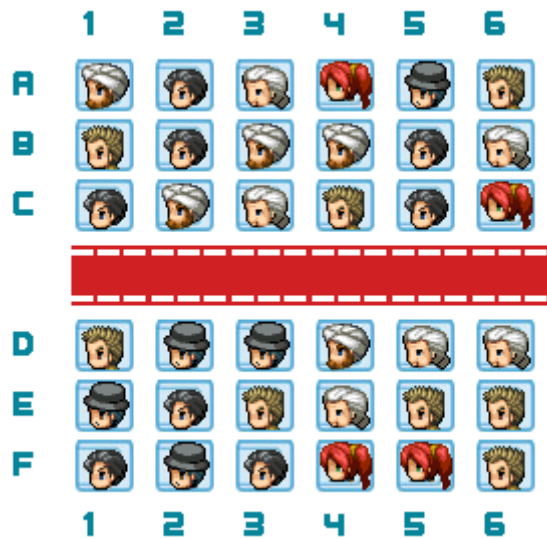

```
(define (problem airbus-seats-p02-6f-36p)
(:domain airbus-seats)
(:objects
    row1 row2 row3 row4 row5 row6 - row
    pass1 pass2 … pass35 pass36 -
passenger)
(:init )
(:goal
    (and
    (occupied row1 A)
    (occupied row1 B)
    ...
    (occupied row6 D)
    (occupied row6 E)
    (occupied row6 F)
    )
))
```

Subir pasajeros a un avión como un problema de planificación automática

Problema 6:
10 filas, 15 pasajeros

Meta

Inicial

Subir pasajeros a un avión como un problema de planificación automática

Problema 6:

```
(:init
    (boarded pass1)   (boarded pass2)
    (boarded pass3)   (boarded pass4)
    (boarded pass5)   (boarded pass6)
    (occupied row1 B)
    (passenger-seated row1 B pass1)
    (occupied row1 C)
    (passenger-seated row1 C pass2)
    (occupied row2 D)
    (passenger-seated row2 D pass3)
    (occupied row2 E)
    (passenger-seated row2 E pass4)
    (occupied row3 B)
    (passenger-seated row3 B pass5)
    (occupied row3 C)
    (passenger-seated row3 C pass6)
)
```

```
(:goal
    (and
    (occupied row1 A)
    (occupied row1 B)
    (occupied row1 C)
    (occupied row2 D)
    (occupied row2 E)
    (occupied row2 F)
    (occupied row3 A)
    (occupied row3 B)
    (occupied row3 C)
    (occupied row4 D)
    (occupied row4 E)
    (occupied row4 F)
    (occupied row5 A)
    (occupied row5 B)
    (occupied row5 C)
    )
)
```

Subir pasajeros a un avión como un problema de planificación automática

Problema 9:
10 filas, 15 pasajeros
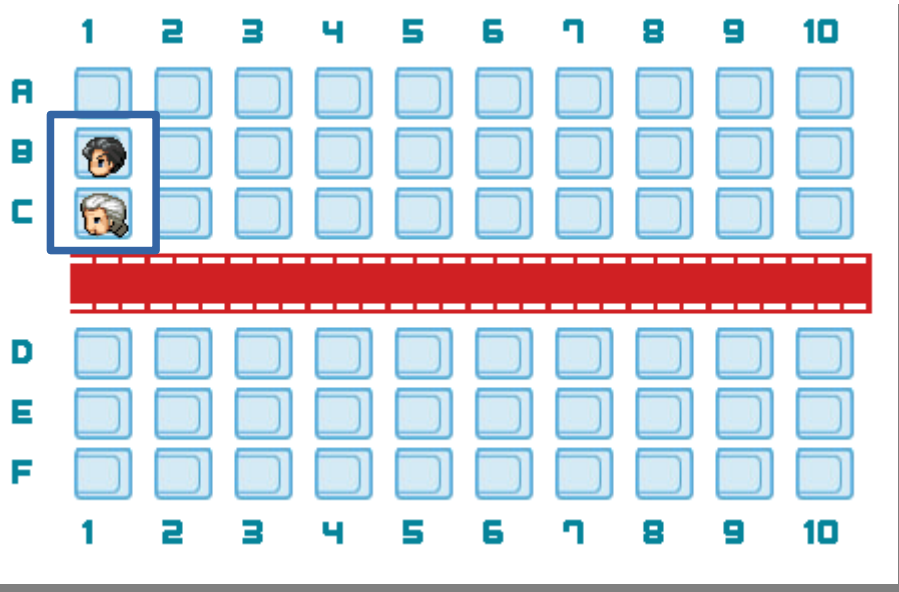
Inicial
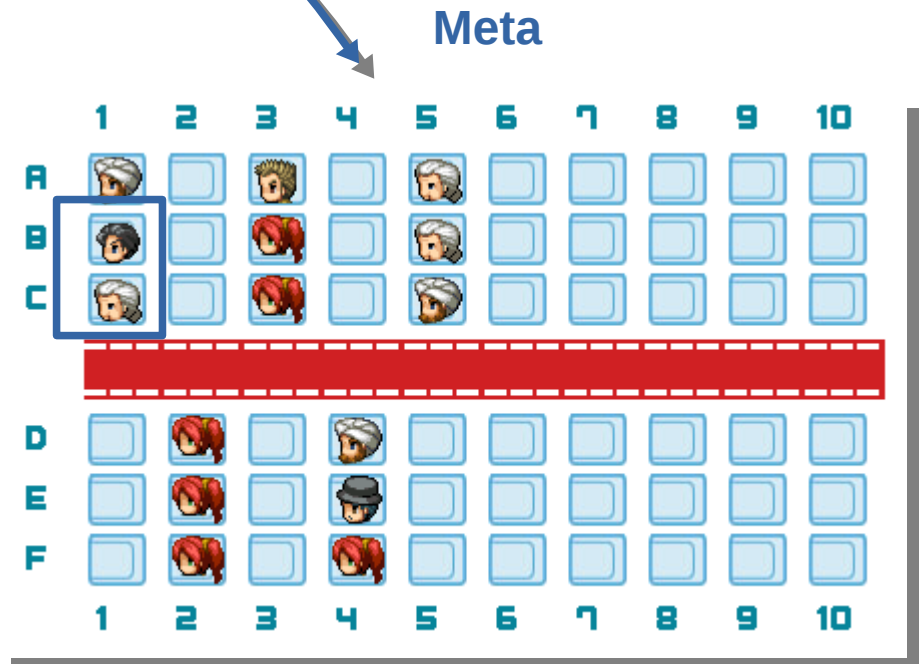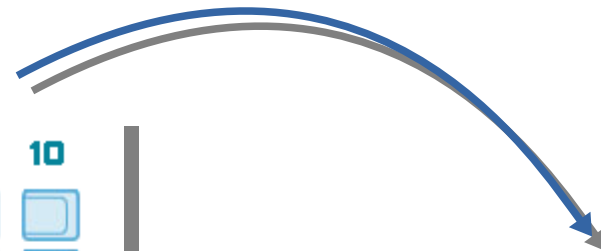
Meta

Subir pasajeros a un avión como un problema de planificación automática

Problema 9:

```
(:init
    (boarded pass1)
    (boarded pass2)
    (occupied row1 B)
    (passenger-seated row1 B pass1)
    (occupied row1 C)
    (passenger-seated row1 C pass2)
)
```

```
(:goal
    (and
    (occupied row1 A)
    (occupied row1 B)
    (passenger-seated row1 B pass1)
    (occupied row1 C)
    (passenger-seated row1 C pass2)
    (occupied row2 D)
    (occupied row2 E)
    (occupied row2 F)
    (occupied row3 A)
    (occupied row3 B)
    (occupied row3 C)
    (occupied row4 D)
    (occupied row4 E)
    (occupied row4 F)
    (occupied row5 A)
    (occupied row5 B)
    (occupied row5 C)
    )
)
```

Resultados:

Problema 1:
6 filas, 6 pasajeros

```
walk-to-row row2 pass1
sit_in_a row2 pass1
walk-to-row row2 pass2
sit_in_b row2 pass2
walk-to-row row2 pass3
sit_in_f row2 pass3
walk-to-row row2 pass4
sit_in_e row2 pass4
walk-to-row row2 pass5
sit_in_c row2 pass5
walk-to-row row2 pass6
sit_in_d row2 pass6
```

Resultados y conclusiones:

Problema	Filas	Pasajeros	Estado Inicial	Meta	Lama-First		MpC (Parallel/Seq)	
					Tiempo	Pasos	Tiempo	Pasos
Problema 1	6	6		2 (ABCDEF)	0.1s	12	0.09s / 1.13s	26 (14) / 14 (14)
Problema 2	6	36		Avión lleno	32.94s	97	-- / --	-- / --
Problema 3	30	6		2 (ABCDEF)	1.78s	12	2.15s / 1.76s	29 (20) / 14 (14)
Problema 4	2	60		Avión lleno	0.16s	24	3.62s / --	64 (40) / --
Problema 5	30	174		Avión lleno*	--	--	-- / --	-- / --
Problema 6	10	15	1(BC) - 2(DE) - 3(BC)	1.3.5(ABC) - 2.4(DEF)	7.4s	52	10.24s / --	150 (28) / --
Problema 7	10	15	1(BC) - 2(DE) - 3(BC)	1.3.5(ABC) - 2.4(DEF)**	--	--	-- / --	-- / --
Problema 8	10	15	1(BC) - 2(DE)	1.3.5(ABC) - 2.4(DEF)**	--	--	-- / --	-- / --
Problema 9	10	15	1(BC)	1.3.5(ABC) - 2.4(DEF)**	11.58s	50	18.56s / --	59 (10) / --

* Las primeras 3 filas son primera clase (Solo 4 asientos por fila: ACDF)

** Los pasajeros mantienen su misma posición en el estado meta.

Muchas gracias