

---

---

# PROGRAMACIÓN AVANZADA

## Práctica de Laboratorio PEC6L - Parte 1

### Convocatoria Ordinaria – Mayo 2016

---

---

*Publicado el 29 de marzo de 2016*

### **Simulación del funcionamiento de dos ascensores en un rascacielos**

Se pretende simular el comportamiento de **dos ascensores** existentes en un rascacielos de 21 plantas (20 plantas más la planta baja), y de las personas que los utilizan.

La implementación del sistema considerará a los ascensores como un recurso compartido que quieren usar concurrentemente varios hilos, que son las personas. Cada ascensor se representará mediante su número de piso actual y su sentido: parado (P), bajando (B), subiendo (S), estropeado (E).

El comportamiento del sistema a tener en cuenta es:

- Cada ascensor tiene capacidad para albergar a un máximo de 8 personas a la vez.
- Por motivos de ahorro energético, solamente hay un ascensor funcionando en cada momento.
- El ascensor que está en funcionamiento está parado mientras nadie lo está utilizando.
- Cada persona es identificada con una “P” más un número (id) único. Ejemplo: P1, P2, P3, etc.
- Las personas se ubicarán inicialmente en las diferentes plantas de manera aleatoria, de tal forma que no todas las personas comiencen en la planta baja. Su destino (la planta a la que se dirigen), también será aleatorio.
- Las personas irán llegando a los ascensores de forma escalonada (no todas a la vez), con un ritmo aleatorio entre ellas de mínimo 0.5 segundos y máximo 2 segundos.
- Cuando una persona quiere utilizar el ascensor debe pulsar el botón del piso en el que se encuentra. Estas pulsaciones quedan registradas hasta que el ascensor las atiende. En cada piso pueden estar varias personas esperando al ascensor.
- En cada planta existe un único botón para llamar a cualquiera de los dos ascensores (se llama al ascensor que esté funcionando en ese momento).

- Cuando una persona sube al ascensor, selecciona el botón del piso de destino. Estas pulsaciones quedan registradas hasta que el ascensor las atiende.
- Cuando el ascensor llega a un piso, todas las personas con destino en ese piso se bajan; y si hay una persona esperando para subir en ese piso, sube y pulsa su destino.
- El ascensor sigue la política de no cambiar de sentido cuando ha iniciado el movimiento, hasta que no quedan peticiones que atender en ese sentido, tanto del interior de la cabina como del exterior.
- El ascensor consume 0'5 segundos en subir o bajar un piso.
- Cada cierto tiempo (aleatorio entre 5 y 7 segundos), el ascensor que está funcionando se estropea, y entonces el segundo ascensor entrará en funcionamiento. Lo mismo ocurre a la inversa: si es el segundo ascensor el que está funcionando y se estropea, el primero volverá a funcionar porque ya estará reparado.
- Cada vez que se estropea el ascensor, las personas que estén dentro se bajan en la planta en la que se ha estropeado y cambian de ascensor (tienen que volver a pulsar el botón de llamada del ascensor). Por simplificar, el ascensor nunca se estropeará entre dos plantas, sino en una planta determinada.
- Cuando un ascensor se estropea, quedará parado en esa planta hasta que esté reparado y vuelva a estar en funcionamiento porque el otro se haya estropeado. Es decir, cuando vuelva a entrar en funcionamiento lo hará desde la planta en la que se estropeó.

Para seguir la evolución del sistema será preciso que se imprima el estado cada vez que el ascensor cambie de posición. El estado del sistema tendrá la siguiente forma:

Piso:	Asc. 1:	Asc. 2:	Botón pulsado:	Destinos del interior:
20			No	-
19			--Sí	-
18		E	No	-
17			--Sí	-
16			No	-
15			No	- P1
14			No	-
13			No	-
12			--Sí	-
11			No	-
10			No	- P2, P3
9			No	-
8			--Sí	-
7			No	-
6			No	-
5			No	-
4			--Sí	-
3			--Sí	-
2			No	-
1	B#4		No	-
0			No	- P4

Esta descripción será **mostrada mediante impresiones por pantalla** con `System.out.println(“”).` El significado del estado anterior es que el ascensor 1 está en la planta 1, bajando (“B”) con 4 personas dentro (“#4”), que en las plantas 3, 4, 8, 12, 17 y 19 se ha pulsado el botón de llamada al ascensor y que las personas que están en el ascensor quieren ir a las plantas 0, 10 (a ésta dos personas) y 15. Por su parte, el ascensor 2 está estropeado en la planta 18.

El comportamiento de las personas se generará aleatoriamente mediante las funciones `random` de Java, y **todo el comportamiento del sistema se guardará en un fichero de texto** (además de mostrarse en pantalla) de forma que sea sencillo analizar lo sucedido. En dicho fichero, entre cada muestra del estado del ascensor se incluirán líneas que representen cada acción de pulsación que se ha producido por las personas que llaman al ascensor o que están dentro del mismo, así como los momentos en los que el ascensor se estropea y la gente que estaba dentro se baja del mismo.

El sistema ejecutará 100 movimientos de ascensor (en total entre ambos ascensores) y creará 60 personas **de forma escalonada y aleatoria**, no todas a la vez. Una vez terminados los 100 movimientos, se mostrará el estado final del ascensor en funcionamiento (personas en su interior), sus destinos, la situación de cada ascensor (estado y planta en la que se encuentra) y las personas que no fueron atendidas. Esta situación también se grabará en el fichero.

El nombre del fichero será “`evoluciónAscensor.txt`”.

Se podrán utilizar todos los mecanismos vistos en clase para resolver todos los problemas de comunicación y sincronización que se plantean en este enunciado.

---

## Condiciones de entrega

1. *La práctica se realizará **individualmente** y deberá ser entregada antes de la fecha indicada en el Aula Virtual, a través de la tarea correspondiente, mediante la subida de dos archivos: la memoria de la práctica en formato PDF o DOC y el proyecto Netbeans completo, comprimido como ZIP o RAR. No se aceptarán trabajos enviados pasada la fecha límite de entrega.*
2. *La entrega fuera del plazo indicado en el Aula Virtual supondrá una reducción en la calificación final, siendo del 25% si se entrega el día siguiente a la fecha límite, o del 50% si se entrega dentro de los dos días siguientes. La entrega más allá de esos dos días no será admitida bajo ninguna circunstancia.*
3. *Es condición necesaria para aprobar, que todos los programas funcionen correctamente y de acuerdo a las especificaciones indicadas en los enunciados.*
4. *Es condición necesaria para aprobar, que todos los nombres de las clases comiencen por una letra mayúscula y todos los nombres de atributos y métodos comiencen por una letra minúscula.*

5. *En la portada de la memoria deberán figurar los datos siguientes:*
  - a. **Grado en Ingeniería \_\_\_\_\_ (Informática o de Computadores)**
  - b. **Curso 2015/2016 – Convocatoria Ordinaria**
  - c. **DNI – Apellidos, Nombre**
6. *La memoria explicativa de la práctica realizada deberá incluir, en el orden siguiente: 1) un análisis de alto nivel; 2) diseño general del sistema y de las herramientas de sincronización utilizados en cada lugar del rascacielos; 3) las clases principales que intervienen con su descripción (atributos y métodos); y 4) un diagrama de clases que muestren cómo están relacionadas.*
7. *Dicha documentación, exceptuando el código, no deberá extenderse más de 20 páginas. La calidad de la documentación – presentación, estructura, contenido, redacción – será un elemento básico en la evaluación de la práctica.*
8. *Para la defensa de la práctica, deberá presentarse una copia en papel de la memoria, impresa por las dos caras y grapada. Este documento podrá ser utilizado por el estudiante como base para responder a las cuestiones que se le planteen en el ejercicio escrito sobre la realización de la aplicación.*
9. *Para mostrar el funcionamiento de los programas, es conveniente que cada estudiante utilice su propio ordenador portátil, en previsión de posibles problemas al instalarlos en alguno de los ordenadores del laboratorio.*