

ENTREGA E3 (0,3 puntos)

Antes de empezar...

- Descargar el archivo **E3.zip** e importarlo a Eclipse.
- Renombrar el proyecto (tecla F2). Por ejemplo, para el grupo formado por Ana Pérez, Jon Azkue y Miren Landa el nuevo nombre sería: **E3_APerezJAzkueMLanda**.
- En el fichero **componentesGrupo**, escribir los nombres de los componentes del grupo.

Además, tened en cuenta que:

- Se valorará la eficiencia de las soluciones, es decir, además de que sean correctas, deben ser eficientes.
- Aparte de los casos de prueba que se os entregan, se espera que incorporéis casos de prueba adicionales y significativos.
- Para entregarlo, debéis exportar el proyecto y subirlo a eGela.

Fecha límite de entrega: 17/10/2023 a las 23:59

—

Ayuda para importar:

File -> Import... -> General -> Existing Projects into Workspace -> Select archive file (el .zip descargado) -> Finish

Ayuda para exportar:

Pinchar en el proyecto. File -> Export... -> General -> Archive File -> (seleccionar las carpetas/archivos a exportar) -> To archive file (escribir una ruta y nombre para el nuevo archivo .zip) -> Finish

Ejercicio 1 (se recomienda leer bien el ejercicio completo antes de comenzar a hacerlo)

Queremos simular el funcionamiento de un determinado parking público. Concretamente, vamos a tener en cuenta dos tipos de eventos: entrada de coches y salida de coches. Para los eventos de entrada conocemos la matrícula del coche. Para los de salida, el nº de plaza que se vacía.

```
public class Evento {  
  
    private Character tipo; // 'E' entrada, 'S' salida  
  
    private String matric; // Matrícula coche (solo si tipo=='E', si no será null)  
  
    private Integer numPlaza; // Plaza que se vacía (solo si tipo=='S', si no será null)  
  
    ...  
  
}
```

En cada plaza de aparcamiento, queremos registrar las matrículas de los coches que han ocupado dicha plaza. Por tanto, cuando un coche entra al parking, su matrícula se registrará en la plaza de aparcamiento que ocupe.

```
public class Plaza {  
  
    private XXXXXXXX registroMatriculas; // TO DO  
  
    public Plaza() { // TO DO }  
  
    public void registrarMatricula(String matricula) { // TO DO }  
  
    ...  
  
}
```

El parking dispone de 50 plazas, numeradas del 0 al 49. En este parking, a cada coche que llega una máquina le reparte un ticket en el que se le indica a qué plaza de aparcamiento debe dirigirse.

```
public class Ticket {  
  
    private int numPlaza;  
    public Ticket(int numPlaza) { this.numPlaza = numPlaza; }  
  
}
```

Consideraremos que al empezar el día el parking está vacío y que cada día se introduce en la máquina una colección de 50 tickets, uno por cada plaza, de manera que el primer ticket a repartir sea el ticket de la plaza 0, luego el de la 1, luego el de la 2, etc. Cuando un coche sale del parking, la máquina genera un nuevo ticket con el número de la plaza que se acaba de liberar y sitúa dicho ticket en el primer lugar de la colección de tickets a repartir.

```
public class Parking {  
  
    private final int TOTALPLAZAS = 50;  
  
    private XXXXXXXX plazas; // TO DO  
  
    private XXXXXXXX ticketsAREpartir; // TO DO  
  
    public Parking() { // TO DO }  
  
    public void simularEventos(XXXXXX eventos){ //TO DO }  
}
```

Se pide:

- (a) En la clase Plaza, completar la definición del atributo *registroMatriculas* y la *constructora* de la clase, que crea un registro de matrículas vacío.
- (b) En la clase Plaza, completar el método *registrarMatricula* (*String matricula*). Este método registra en dicha plaza la llegada del coche con la matrícula indicada.
- (c) En la clase Parking, completar la definición de los atributos *plazas* y *ticketsAREpartir*. Completar también la *constructora* de la clase, que permite crear las plazas y tickets necesarios.
- (d) En la clase Parking, completar el método *simularEventos*(XXXXXX *eventos*). Este método recibe el conjunto de Eventos de entrada y salida que han ocurrido durante una jornada (debes determinar de qué tipo será el atributo *eventos*). El método simulará los eventos de entrada y salida tal y como se ha especificado en el enunciado. Además, al final se mostrará el registro de matrículas de cada plaza.
- (E) En el programa de prueba, completar el tipo de la variable *eventos* (además de añadir otros casos de prueba que se os puedan ocurrir)

Además, es posible que tengáis que definir algunos getters y/o setters.

Ejemplo de simulación: (para los eventos de tipo E se muestra la matrícula del coche que entra, para los de tipo S el nº de plaza que se libera)

Eventos:

E 1212ABS
E 3412GHT
E 7564ODE
E 9876YYY
S 2
E 5902BHE
S 2
S 1
E 6112DEX
E 7456MKS
E 9999ABC



Pantalla:

---- Simulación de eventos ----
El coche 1212ABS va a la plaza 0
El coche 3412GHT va a la plaza 1
El coche 7564ODE va a la plaza 2
El coche 9876YYY va a la plaza 3
Se ha liberado la plaza 2
El coche 5902BHE va a la plaza 2
Se ha liberado la plaza 2
Se ha liberado la plaza 1
El coche 6112DEX va a la plaza 1
El coche 7456MKS va a la plaza 2
El coche 9999ABC va a la plaza 4

---- Registro de matrículas ----
Plaza 0: 1212ABS
Plaza 1: 3412GHT 6112DEX
Plaza 2: 7564ODE 5902BHE 7456MKS
Plaza 3: 9876YYY
Plaza 4: 9999ABC
...