

	Ciclo Formativo de Grado Superior. Desarrollo de Aplicaciones Multiplataforma / Desarrollo de Aplicaciones Web
	Módulo: Programación
	Profesor: Á. Enrique Pineda Navas
	Examen 3ª Evaluación
Fecha: 26/05/2022	Nombre:

1. ESTRUCTURA Y CARACTERÍSTICAS

- La prueba se basa en los criterios de evaluación que establece el currículo vigente del ciclo formativo de grado superior de Técnico Superior en Desarrollo de Aplicaciones Multiplataforma en la Comunidad Autónoma Extremadura del módulo PROGRAMACIÓN (DECRETO 259/2011 - Código: 0485)
- Consta de **3 ejercicios**, cuya puntuación se detalla en el enunciado de cada uno de ellos, pudiendo alcanzar un máximo de **10** puntos en total. La nota de esta prueba representará el 60% de la nota del módulo.
- La duración total de la prueba es de **4 horas**, pudiendo el alumno organizar el orden de realización de los ejercicios como considere oportuno. A modo de orientación, el tiempo estimado en la realización de los ejercicios es de 30' para el ejercicio 1 y 1 hora para los ejercicios 2 y 3.

2. NORMATIVA

- La prueba se realizará utilizando el lenguaje de programación **JAVA** y el entorno de desarrollo **Netbeans**.
- NO está permitido** la utilización de Internet ni tampoco del material desarrollado durante el curso (apuntes y prácticas), dado que los ejercicios planteados en esta prueba son similares a las prácticas realizadas en clase.
- NO está permitido** el uso del teléfono móvil (se recomienda que esté apagado) ni servicios de correo electrónico, chats o demás programas de mensajería, debido al carácter **INDIVIDUAL** de la prueba.
- Se valorará no solo el resultado del ejercicio sino también la eficiencia del código implementado por el alumno.
- La penalización por falta de claridad en el código y/o mala utilización o ausencia de sangría será del 10% de la nota del apartado.

3. EJERCICIOS

IMPORTANTE: PARA QUE UN EJERCICIO SEA VALORADO ES IMPRESCINDIBLE QUE COMPILE Y SE EJECUTE.

	Ciclo Formativo de Grado Superior. Desarrollo de Aplicaciones Multiplataforma / Desarrollo de Aplicaciones Web
	Módulo: Programación
	Profesor: Á. Enrique Pineda Navas
	Examen 3ª Evaluación
Fecha: 26/05/2022	Nombre:

EJERCICIO 1.- (4 puntos) – FICHEROS DE OBJETOS (Mínimo 1 punto) – El aumento de los precios de los coches, tanto nuevos como de segunda mano, motivado por la combinación de varios factores como la falta de stock en los concesionarios o la crisis de materiales semiconductores, han llevado al concesionario *Car-Run* a realizar un importante desembolso en un software de gestión. Este software almacenará la información de los coches que se encuentren en el concesionario. De cada coche almacenaremos su matrícula, kms y la fecha de matriculación (Utilizando las clases específicas para ello).

Al inicio de la ejecución, nuestro programa cargará todos los coches del concesionario (almacenados en el fichero *carrun.obj*) en un HashSet **(0,75 puntos)** para, inmediatamente después, mostrar un menú al usuario con las siguientes opciones:

1. Insertar coche en el conjunto. **(0,5 puntos)**
2. Poner a cero los km (Pedirá un año al usuario y pondrá a 0 el atributo km de todos los coches del conjunto cuya fecha de matriculación coincida con el año introducido por el usuario). Utiliza un for-each para ello. **(1 punto)**
3. Buscar por matrícula (en el conjunto). **(1 punto)**

Una vez el usuario salga de nuestro programa, debemos volcar todos los coches al fichero **(0,75 puntos)**.

EJERCICIO 2.- (4 puntos) – ESTRUCTURAS DE DATOS Y FICHEROS DE TEXTO (Mínimo 1 punto) – La empresa de ambulancias *Los ángeles da saúde* desean registrar las llamadas a éstas para poder realizar la toma de decisiones en su consejo de administración en base a datos. Es por ello que les ayudaremos desarrollando un software.

Debemos almacenar la información de las ambulancias en un vector de 4 posiciones. Para cada ambulancia, tendremos su matrícula y un HashMap de llamadas, donde cada *llamada* almacenará la fecha de la llamada (la de hoy) y el nombre del paciente que la ambulancia debe atender.

1. Llegada de ambulancia. (Insertaremos una ambulancia -sin ninguna llamada- en el primer hueco libre del vector. Si no hay hueco libre, muestra un mensaje de aviso). **(0,75 puntos)**
2. Registrar llamada (Pediremos el nombre del paciente y, posteriormente, la posición en el vector de la ambulancia que lo atenderá. Si la posición está vacía, la pediremos hasta que nos dé una ocupada. Una vez nos dé una posición ocupada, insertaremos la llamada en el HashMap de la ambulancia) **(0,5 puntos)**
3. Buscar paciente (Pediremos la clave del paciente y mostraremos la matrícula de la/s ambulancia/s que atienden un paciente con dicha clave. Puede haber pacientes con la misma clave en distintas ambulancias) **(1 puntos)**
4. Eliminar ambulancia. Elimina del vector la ambulancia según su matrícula. Si no existe, mostraremos el mensaje “No existe ninguna ambulancia con la matrícula [matrícula]”. **(0,5 puntos)**

	Ciclo Formativo de Grado Superior. Desarrollo de Aplicaciones Multiplataforma / Desarrollo de Aplicaciones Web
	Módulo: Programación
	Profesor: Á. Enrique Pineda Navas
	Examen 3ª Evaluación
Fecha: 26/05/2022	Nombre:

5. Salir del programa.

Hasta que el usuario no pulse 5 no saldremos del programa y se volverá a mostrar el menú. Cuando el usuario pulse la opción 5, se escribirán en el fichero *pacientes.txt* todos los pacientes que haya en ese momento en la estructura de datos. **(1 punto)**

Ejemplo de fichero:

```
2022-05-26#Justo#Jus
2022-05-26#Joaquina#Joa
2022-05-26#Paco#Pac
2022-05-26#María Antonia#Mar
```

NOTAS: La clave de los mapas serán las tres primeras letras de los nombres de los pacientes. Controlaremos, mediante la excepción correspondiente, que la opción del menú no sea una letra. La pediremos una y otra vez mientras no sea un número. **(0,25 puntos)**

EJERCICIO 3.- (2 puntos) – ARRAYLIST (Mínimo 0,5 puntos) – Realiza un programa en el que utilices una lista de enteros y que muestre un menú en el que se le ofrezcan al usuario las siguientes opciones:

1. Rellenar, con números aleatorios entre 1 y 100, las ventas de coches de cada uno de los 12 meses del año. **(0,5 puntos)**
2. Mostrar las ventas introducidas en el punto anterior. **(0,25 puntos)**
3. Que muestre las ventas totales de los meses cuyo nombre contenga la letra a. **(0,5 puntos)**
4. Que muestre el nombre del mes (o meses) con más ventas. **(0,75 puntos)**
5. Salir del programa.

Controlaremos que el usuario elija una opción del menú entre 1 y 5. Hasta que el usuario no pulse 5 no saldremos del programa.

PISTA: Necesitarás, un array unidimensional con el nombre de los meses del año.