

Módulo: Programación			
Evaluación: Tercera	Hora comienzo: 17:00	Hora Finalización:21:00	Fecha: 27/05/2024

Normas para la realización del examen
Todos los ejercicios se guardarán en un proyecto perfectamente empaquetados.
El proyecto se comprimirá y el archivo zip/rar resultante llevará el siguiente nombre: Nombre_Apellido1_Apellido2
Cada ejercicio se identificará por su número correspondiente en el examen.
Los ejercicios deberán funcionar correctamente.
Llegada la hora de recogida su subirá a la plataforma
Todos los ejercicios deberán estar bien empaquetados (package) así como deberán cumplir con CamelCase y otros estilos de programación ya empleados en clase, tabulaciones, comentarios, etc.

## Ejercicio 1:

Implementar una aplicación para gestionar las existencias de una tienda de repuestos de automóviles.

Cada producto se identificará por un código alfanumérico. La aplicación permitirá dar de alta o de baja productos y actualizar el número de unidades en stock de cada uno de ellos. Los datos se mantendrán en un fichero, que deberá actualizarse al cerrar el programa. Emplea colecciones. (2,5ptos)

## Ejercicio 2:

Crear una lista de números enteros positivos introducidos por consola hasta que se introduzca uno negativo. A continuación, recorrer la lista y mostrar por pantalla los índices de los elementos de valor par, que será multiplicado por 100. (1,5pto)

Ejemplo:

[2, 4, 6, 8, 10, 12, 15, 14]

Índices de valores pares:

0

1

2

3

4

5

7

[200, 400, 600, 800, 1000, 1200, 15, 1400]

### Ejercicio 3:

Crea una clase cuenta corriente que tenga los siguientes atributos, DNI, Nombre, Apellido, Saldo, los tipos que tú consideres más adecuados. Escribe un programa en Java que crea un ArrayList de cuentas corrientes. El programa pedirá por teclado los datos de las cuentas y los guardará en el array. A continuación, utilizará el ArrayList para mostrar por pantalla los siguientes métodos: (2,5ptos)

- a) Todas las cuentas corrientes.
- b) Todas las cuentas de un determinado usuario, este se pedirá por teclado.
- c) Todas las cuentas que tengan un saldo inferior a uno que se introduzca por teclado.
- d) Todas las cuentas corrientes ordenadas por saldo de menor a mayor.

### Ejercicio 4:

Realizar un programa que lea una frase del teclado y nos indique si es palíndroma, es decir, que la frase sea igual leyendo de izquierda a derecha que de derecha a izquierda. Sin tener en cuenta los espacios. Un ejemplo de palíndroma es: “Dábale arroz a la zorra el abad”. Las vocales con tilde hacer que los algoritmos consideren una frase palíndroma como si no lo fuese. Por esto, supondremos que el usuario introduce la frase sin tildes. (1pto)

### Ejercicio 5:

Un distribuidor de productos fitosanitarios decide proponer a un equipo de programadores el diseño de una página web en la que los usuarios realizarán los diferentes pedidos. Para ello será necesario crear una clase llamada Pedidos. A Pedidos se irán insertando los diferentes elementos fitosanitarios que se guardarán en una lista. Tendrás que crear una lista Fitosanitarios. Cada fitosanitarios que tengas en el pedido deberá contener su nombre, precio y las unidades de cada uno. Si se inserta en la lista un producto que ya existe se incrementará su cantidad, pero no se repetirá. Métodos que tendrás que implementar en pedidos: (2,5ptos)

- a) Número de elementos del pedido
- b) Importe total del pedido



Salida del programa:

Contenido del pedido:

Insecticida Malas pulgas PVP 19,33 Unidades : 2 Subtotal 39,90

Herbicida Radical PVP 44,33 Unidades : 1 Subtotal 44,33

Bactericida Battion. PVP 19,95 Unidades : 1 Subtotal 19,95

En tu pedido tienes 3 productos

El total del pedido es de 105,18 Euros.

Ejercicio 1	Sin fichero 1,5 ptos	Con fichero 1 pto
Ejercicio 2	Crear lista: 0,5ptos Mostrar números pares: 0,5 ptos Multiplicado por 100: 0,5 ptos	
Ejercicio 3	Clase 0,5 ptos ArrayList 0,5 ptos Petición por teclado 0,5	Apartados a,b,c,d : 0,2 ptos por apartado
Ejercicio 4	1 pto	
Ejercicio 5	Clase : 0,5 ptos Entrada datos: 0,5 ptos Colecciones HashMap: 0,5 ptos Salida por pantalla 1 pto	