

# UD07.EXAME Práctico

DAM1-Programación 2023-24

08/05/2024

1. Login y Log de Accesos (5)

2

2. Máximas puntuaciones (5)

3

- Puedes utilizar apuntes y materiales que consideres pero deberás realizar los programas individualmente. En caso contrario se retirará el examen.
- Realiza programas bien estructurados, legibles, con comentarios, líneas en blanco, identificadores adecuados, etc.
- Cuida la interacción con el usuario, presentando la información de forma clara y ordenada.
- Nombra los proyectos y ficheros con nombres adecuados.
- Indica la autoría del código incluyendo un comentario con tu nombre y apellidos.

1. Descarga el proyecto Java del **repositorio** (o donde te indique el profesor) y configúralo como un nuevo proyecto Java en tu entorno de desarrollo.
2. Crea el código fuente en los paquetes que se indican en el enunciado.
3. **Incluye un comentario con tu nombre y apellidos al inicio de cada clase.**
4. **Comprime y entrega la carpeta /src** del proyecto comprimida.
5. **Tiempo máximo: 2:20 horas**

# 1. Login y Log de Accesos (5)

Paquete: **loginlog**

Descarga el siguiente proyecto Java (vscode).

La clase `CifradoBcrypt` contiene los siguientes métodos estáticos:

1. `generarHash2Y()`: Genera un hash de BCrypt versión 2Y a partir de una contraseña en texto plano en formato String
2. `validarHash2Y()`: Valida un hash de BCrypt versión 2Y con una contraseña en texto plano en formato String

La carpeta `lib` contiene las librerías necesarias para hacer funcionar los métodos anteriores.

El fichero de texto `shadow.txt` contiene una relación de los nombres de los usuarios de una aplicación y sus contraseñas cifradas mediante el algoritmo Bcrypt, con el siguiente formato en cada línea:

```
username: hash2Ypassword
```

La contraseña de cada usuario es igual a su propio nombre de usuario.

Crea una aplicación JavaFX, **AppLogin.java**, que nada más empezar solicite un nombre de usuario y contraseña y lo valide contra los datos del fichero.

Si las credenciales son erróneas mostrará un mensaje de error sin especificar si es el usuario o la contraseña lo que falla (ya que dar esa información podría ser útil a posibles atacantes). El usuario podrá cerrar el mensaje e intentarlo de nuevo.

Si las credenciales son correctas se cerrará la ventana inicial y se mostrará otra ventana con el texto "Bienvenid@" tanto en el título como en el centro de la ventana.

Tanto si las credenciales son válidas como si no, se registrará el intento de inicio de sesión en un fichero de texto de nombre `access.log`, ubicado en el mismo directorio que el fichero `shadow.txt`. Se añadirá al fichero `access.log` una línea de texto indicando la fecha, la hora, el nombre de usuario y si el acceso ha tenido éxito o no, usando el siguiente formato:

```
aaaa-mm-dd hh:mm:ss nombre_de_usuario {OK|ERR}
```

Si el fichero `access.log` no existe, se creará. Si existe, se añadirá la línea al final.

## 2. Máximas puntuaciones (5)

Paquete: **puntos**

Se desea implementar una funcionalidad para guardar en un fichero binario las máximas puntuaciones de las personas que juegan a un juego.

Para ello:

1. Crea una nueva clase `Puntuacion` que permita almacenar el nombre del jugador o jugadora, la puntuación obtenida y la fecha y hora en que la consiguió.
2. Crea un método que guarde una lista de puntuaciones en un fichero binario cuyo nombre también se pasa como parámetro. Si el fichero no existe deberá crearse.
3. Crea otro método que devuelva una lista de puntuaciones leídas de un fichero binario cuyo nombre se pasa como parámetro.
4. Crea una aplicación, de consola o JavaFX, **AppAddPuntos.java**, para generar puntuaciones que solicite por teclado un nombre de jugador o jugadora y su puntuación y añada la puntuación con la fecha y hora del momento a un fichero de nombre "puntuaciones.dat". Genera al menos 5 puntuaciones.
5. Crea una aplicación JavaFX, **AppShowPuntos.java**, para mostrar las puntuaciones contenidas en un fichero. La aplicación se iniciará solicitando al usuario el fichero que desea abrir y a continuación las mostrará ordenadas de mayor a menor.