

NOTA ACLARATORIA:

El presente examen corresponde a la evaluación final del módulo de PROGRAMACIÓN DE SERVICIOS Y PROCESOS. Para la corrección del mismo ***se valorarán las buenas prácticas de programación y limpieza a la hora de escribir el código, teniendo en cuenta además, la elegancia e ingenio en la resolución de los problemas propuestos y los COMENTARIOS*** tal como se han ido adquiriendo los conocimientos a lo largo del trimestre. **Los apartados/ejercicios Deben funcionar CORRECTAMENTE para ser evaluados!!!**

Hay que obtener un 5 (como mínimo) en este examen **para poder aprobar el módulo**

Entregaréis las clases correspondientes a cada ejercicio en carpetas, para posteriormente entregar un fichero comprimido con vuestro nombre en la forma siguiente:

NombreApellidos_Curso → Ejemplo: **AndresChillonSesma_DAM2.zip** (rar, 7Z, ...)

EJERCICIO 1 (2 PUNTOS)

Dado la siguiente jerarquía de archivos (carpeta adjunta en la tarea de examen comprimida y que copiaréis descomprimida en el directorio de vuestro proyecto (para no tener que acceder por ruta absoluta, sino por ruta relativa en el código de la forma *MiClasePSP/<elementos>*:

MiClasePSP

```

|_____Alumno1
|      |_____Notas *
|
|                      Programación 10
|                      BBDD 5
|_____Alumno2
|      |_____Notas
|
|                      Programación 4
|                      BBDD 7
|_____Alumno3
|      |_____Notas
|
|                      Programación 6
|                      BBDD 6
|_____Alumno4
|      |_____Notas
|
|                      Programación 8
|                      BBDD 9

```

* (Notas es un fichero donde aparecen las notas de las asignaturas)

Crear un fichero de Asignaturas y dejarlo en C:/Examen con el nombre **"MediasAsignaturas_<VuestroNombre>"** que contenga la media de cada asignatura, de la forma:

Asignatura, NotaMedia

Programación, 6.5

BBDD, 6.75

IMPORTANTE: Las asignaturas de cada alumno pueden ser diferentes, Es por ello que tendréis que controlar que pueda haber asignaturas distintas. NO deis por hecho que solamente están esas dos del ejemplo.

Podéis utilizar las clases que consideréis del tema de ficheros vistas, comentando el código lo justo para saber qué es lo que estáis haciendo de cara a una mejor comprensión

EJERCICIO 2 (1 PUNTO)

Con el fin de realizar un aprovisionamiento de clientes de confianza, en la empresa ACME han decidido iniciar un proceso de validación y verificación de clientes a través de la información fiel y veraz de los mismos. Para ello tendremos que realizar un programa en el que tengamos un servidor a la escucha de clientes que se quieran conectar (1..N) y cada cliente que se conecte lo hará con un nombre y recibirá del servidor un fichero CIFRADO.txt (que estará cifrado).

El fichero que recibirá cada cliente, se almacenará en su propia ruta. Nosotros, al estar en el mismo PC, vamos a dejarlo en la ruta relativa `./pendFirma/<nombreCliente>/CIFRADO.txt`

De este modo podremos tener todos los ficheros en el mismo almacén, simulando diferentes ubicaciones remotas.

Una vez el fichero esté en el cliente:

- Se firme con el nombre del hilo (cliente) que se ha conectado de manera automática.
- Se guarde una copia del fichero firmado con el nombre del cliente (txt)
- Se devuelva al servidor firmado.
- En el servidor se guarde ese fichero firmado por el cliente **en una carpeta con su nombre** dentro de un directorio `/firmados`
- Se finalice la conexión con ese cliente.

EJERCICIO 3 (2 PUNTOS)

Tomar como base el ejercicio anterior y realizar una copia del proyecto para realizar exactamente lo mismo pero añadiendo el resumen a todas las comunicaciones entre el servidor y cada uno de los clientes. Recuerda generar el resumen en cada uno de los nodos y posteriormente realizar la operación de validación del resumen en el lado contrario (realizar una verificación para comprobar que dicho fichero es el original que envía el servidor). Date cuenta que lo que queremos guardar es el fichero de manera legible en ambos lados.

EJERCICIO 4 (2 PUNTOS)

Tomar como base el ejercicio anterior e introducir el cifrado con clave privada y pública.

Recuerda generar las claves en cada uno de los nodos e intercambiarlas debidamente. No utilizéis el truco de generarlas solamente en un sitio y, como están albergadas en la misma máquina son legibles desde esa ubicación en cualquiera de los nodos (es trampa). Hay que pensar que las ejecuciones pueden efectuarse en máquinas diferentes.

Nota: recordad que hay que realizar el intercambio de claves, ya que se va a utilizar un sistema de encriptación de clave pública. (Ejercicios 8 y 9)

1. Firmar los datos de un fichero con la clave privada
2. Verificar la firma de un fichero con la clave pública

EJERCICIO 5 (3 PUNTOS)

Realizar una aplicación cliente/servidor en la que cuando un cliente se conecta se le ofrecen dos opciones:

- Acceder a la aplicación (Login)
- Darse de alta en la aplicación

Si el cliente elige la opción de darse de alta se le solicitará un nombre y una password, que se enviará al servidor y éste realizará las comprobaciones a través de un fichero interno.

Si el cliente se encuentra entre los datos de alta y la contraseña es correcta se le enviará desde el servidor un código numérico aleatorio en forma de vale descuento (8 dígitos).

Si el cliente no se encuentra se le responderá desde el servidor y se le vuelve a mostrar las opciones para que pueda volver a iniciar la sesión o darse de alta.

Si el cliente se encuentra pero la contraseña es correcta habrá que indicárselo y pedirle de nuevo la contraseña.

Implementa una solución que de respuesta a esta petición de la manera más elegante.