







LABORATORIO DE SISTEMAS (GIRS)

EXAMEN PRÁCTICO CONVOCATORIA EXTRAORDINARIA - SHELL SCRIPTING, PYTHON SCRIPTING Y GIT

22 DE JUNIO DE 2023

-
-  **Profesores:** Elena García-Morato y Felipe Ortega.
 -  **Lenguaje de programación:** Bash, sed, awk, Python, git.
 -  **Hora tope de entrega:** Jueves, 22 de junio de 2023 a las 11:50.
 -  **Envío:** Código fuente **comentado** a través de Aula Virtual y GitLab de la EIF.
-

INSTRUCCIONES

- Lee con detenimiento el enunciado de cada problema y plantea la solución antes de lanzarte a programar para resolverla.
- Puedes consultar el documento anexo con información de utilidad para resolver los ejercicios. No es posible consultar ninguna página de Internet ni acceder a los contenidos de Aula Virtual durante el examen.
- **Cada *script* debe estar almacenado en un fichero diferente. El nombre de cada fichero debe coincidir exactamente con el indicado en el enunciado del problema.**
- Una vez que termines, envía todos los *scripts* a través del espacio de entrega que se ha habilitado para el examen, en la sección Evaluación. No es necesario comprimir los archivos antes de enviarlos. Sube cada *script* al espacio de envío y luego **confirma el envío** para concluir el examen.
- **IMPORTANTE:** El espacio de envío se cierra automáticamente una vez superada la hora límite de finalización de la prueba. Por favor, no apures hasta el último momento para enviar tus respuestas.

Ejercicio 1

Puntuación: 3 puntos

En el apartado de archivos anexos de este examen práctico en Aula Virtual, podrás encontrar dos archivos de texto necesarios para realizar este ejercicio:

- ▶ El archivo `direcciones.txt` contiene las direcciones de 6 perfiles ficticios.
- ▶ El archivo `correspondencias.txt` indica la relación de sustituciones entre la abreviatura de una comunidad autónoma y su nombre completo.

Escribe un script de *shell* `informe.sh` que toma como entrada el contenido del fichero `direcciones.txt` y utiliza `sed`, `awk` y la información contenida en el fichero `correspondencias.txt` para imprimir por pantalla la lista *ordenada* de comunidades autónomas que aparecen en los datos de entrada, seguida cada una de ellas por el nombre y la ciudad en la que residen.

A continuación, se muestra un ejemplo detallado de la salida que debe producir este *script* para los datos proporcionados:

Comunidad	Nombre y apellidos	Ciudad
Andalucía	Luis López Paz	Granada
Castilla-La Mancha	Elvira López González	Ciudad Real
Comunidad de Madrid	Gloria Duarte Fraile	Madrid
Comunidad Valenciana	Sebastián Sánchez Ruiz	Valencia
Galicia	José Feito Martínez	A Coruña
Región de Murcia	Fulgencio Conesa Bernal	Murcia
País Vasco	Iñaki Satrústegui Olavide	Bilbao

FIN DE LA LISTA

Ejercicio 2

Puntuación: 3 puntos

Crea un *script* en Python llamado `enteros.py` que se ajuste a las siguientes especificaciones:

- ▶ Debe recibir como argumentos *obligatorios* entre 1 y 6 números **enteros**, separados entre sí por un espacio en blanco.
- ▶ Si recibe el parámetro *opcional* `-h` o bien `--help` imprime información de ayuda sobre la sintaxis para ejecutar el *script* y las opciones disponibles.
- ▶ Si recibe el parámetro *opcional* `-z` o bien `--zero` imprime por pantalla el número de ceros que contiene la lista de enteros introducida por el usuario. Ejemplo de ejecución:

```
$ python3 enteros.py --zero 3 0 0 9 30 15
La lista contiene 2 ceros
```

- ▶ Si recibe el parámetro *opcional* `-s` o bien `--sides` devuelve el primer y el último número de la lista. Ejemplo de ejecución:

```
$ python3 enteros.py --sides 3 0 0 9 30 15
El primer número de la lista es el 3 y el último el 15
```

- ▶ Si no se recibe ningún parámetro *opcional*, realiza la suma de los números recibidos como argumento y la imprime por pantalla. Ejemplo de ejecución:

```
$ python3 enteros.py 3 0 0 9 30 15
57
```

- ▶ En caso de que se introduzca algún argumento incorrecto (opción diferente de las aceptadas, números con decimales, argumentos no numéricos, etc.) o bien se introduzca una cantidad de números enteros fuera del rango admitido, el programa deberá concluir la ejecución de inmediato, informando al usuario por pantalla del motivo del error y devolver a la terminal el código de *status* adecuado.

Ejercicio 3

Puntuación: 4 puntos

En la dirección:

`git@gitlab.etsit.urjc.es:egarciap/Hummus.git`

se aloja un repositorio que contiene indicaciones para elaborar un delicioso plato de Hummus. En él podrás encontrar dos archivos: `elaboración.txt`, que contiene los pasos a seguir para su elaboración e `ingredientes.txt`, que contiene los ingredientes necesarios para llevar a cabo la misma.

- ▶ Crea una copia de éste repositorio en tu máquina local y verifica su estado.
- ▶ Partiendo de la única rama existente, crea una nueva rama llamada `spicy`, que contendrá una versión picante de la receta. En ella, añade **al final del fichero `ingredientes.txt`** un nuevo ingrediente: *Pimentón Picante*. Registra los cambios a través de un *commit* con el mensaje “Nuevo ingrediente añadido”
- ▶ En esta misma rama modifica el fichero `elaboracion.txt` añadiendo un nuevo paso al final del mismo: “*Espolvorear el Hummus con pimentón*”. Registra de nuevo dichos cambios a través de un *commit* con un mensaje descriptivo de tu elección.
- ▶ De vuelta en la rama `master`, añade **al final del fichero `elaboración.txt`** un nuevo paso: “*Triturar la mezcla*”, registrando dichos cambios a través de un nuevo *commit*.
- ▶ Darle un toque picante al Hummus nos ha parecido muy buena idea, así que vamos a integrar los cambios llevados a cabo en la rama `spicy` sobre la rama principal, **resolviendo los conflictos que pudieran generarse**.
- ▶ Realiza un último *commit* que indique que ambas ramas han sido fusionadas y, a continuación, etiquétalo (con una etiqueta anotada) como la `v2.0`.
- ▶ Por último, desde la propia terminal de comandos, configura un repositorio remoto para tu repo local llamado `RecetaHummus` en tu cuenta de GitLab de la EIF. Este segundo repositorio remoto se debe identificar en tu copia local por el alias `upstream`. Debes subir todos los cambios realizados y las etiquetas a este repositorio remoto para poder ser evaluado.

Envía a través del espacio de entrega de Aula Virtual un documento de texto plano nombrado `RecetaHummus.txt` que contenga **una línea con la URL *clonable* del repositorio remoto** que has creado.