



# Fundamentos del Software

## GIIADE – A2

*Ejercicio de Clase Sesión 5.2*

### 1 Instrucciones

---

Los ejercicios a resolver deberán estar adecuadamente documentados. Se entregará una memoria con el enunciado del ejercicio en cuestión y a continuación se propondrá la solución. Finalmente, el documento habrá de incluir una sección de pruebas.

También se adjuntarán el(los) script(s) necesario(s) para resolver los ejercicios.

### 2 Documentos a entregar

---

Se deben entregar dos archivos

1. PDF con la memoria.
  2. Archivo ‘.sh’ con el script, guion o las instrucciones necesarias a ejecutar.
- ⚠ En caso de no entregar alguno de estos archivos **no se corregirá** el ejercicio.

### 3 Enunciado

---

Cree un guion que acepte 2 argumentos como parámetro. El primero de ellos será una letra. El segundo el directorio o la ruta de un directorio. El guion se encargará de comprobar que la letra introducida está en minúscula. Y que además el directorio existe.

Además, el guion al introducir una ‘a’ o una ‘e’ debe contar el número de archivos que hay en el directorio que empiezan por esa letra. Por otro lado, si es cualquiera de las otras vocales (i, o, u), entonces que haga una copia de seguridad de los archivos del directorio creando un comprimido ‘.tar’. En caso de existir el archivo, es decir, que ya se ha creado, simplemente que indique que el archivo existe.

### 4 Criterios de evaluación

---

- Se entregan los dos ficheros.
- El script debe de funcionar sin errores de sintaxis.
- La memoria está documentada (una breve frase para cada cosa es suficiente).
- Hay pruebas en el código.



## 5 Memoria

---

La memoria ha de incluir:

- Nombre de los autores.
- Grupo.
- Curso.
- Titulación.

En la resolución de cada ejercicio:

- Enunciado.
- Guion, script o solución para dicho ejercicio.
  - o Esto irá duplicado. Copiado tanto en la memoria como un archivo separado para poder ejecutarse.
- Capturas de pantalla de varias pruebas.

Un ejemplo de guion se muestra a continuación. Observad que en el ejemplo hay:

- ⚠ Comentarios.
- ⚠ Y en las capturas aparece el nombre.

## 6 Plazo

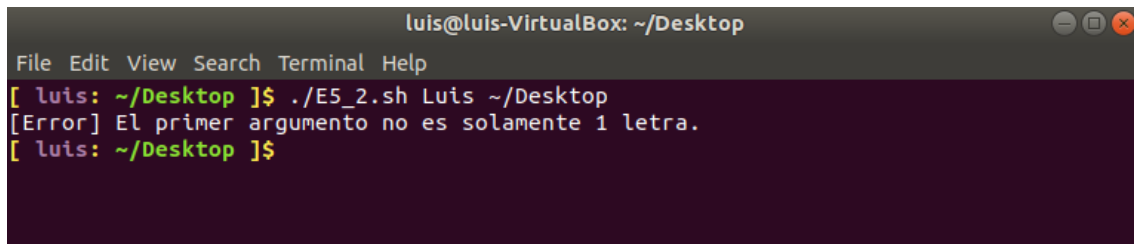
---

Los ejercicios se podrán entregar hasta el mismo jueves en el horario habitual.

## 7 Ejemplos de salidas

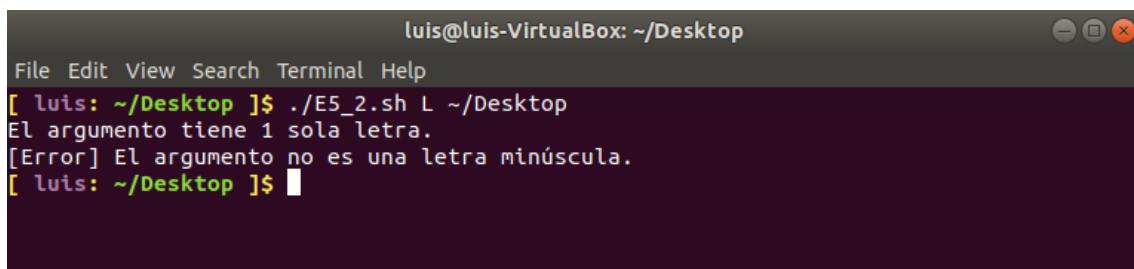
---

Más de una letra en el primer argumento.



```
luis@luis-VirtualBox: ~/Desktop
File Edit View Search Terminal Help
[ luis: ~/Desktop ]$ ./E5_2.sh Luis ~/Desktop
[Error] El primer argumento no es solamente 1 letra.
[ luis: ~/Desktop ]$
```

Una letra mayúscula en el primer argumento.



```
luis@luis-VirtualBox: ~/Desktop
File Edit View Search Terminal Help
[ luis: ~/Desktop ]$ ./E5_2.sh L ~/Desktop
El argumento tiene 1 sola letra.
[Error] El argumento no es una letra minúscula.
[ luis: ~/Desktop ]$
```



Una letra minúscula, no vocal y un directorio existente.

```
luis@luis-VirtualBox: ~/Desktop
File Edit View Search Terminal Help
[ luis: ~/Desktop ]$ ./E5_2.sh l ~/Desktop
El argumento tiene 1 sola letra.
El argumento está en minúscula.
El directorio existe.
La letra no es una vocal. No se hará nada.
[ luis: ~/Desktop ]$
```

Vocal en minúscula 'a' y directorio existente.

```
luis@luis-VirtualBox: ~/Desktop
File Edit View Search Terminal Help
[ luis: ~/Desktop ]$ ./E5_2.sh a ~/Desktop
El argumento tiene 1 sola letra.
El argumento está en minúscula.
El directorio existe.
El número de archivos que empiezan con la letra "a" en el directorio "/home/luis/Desktop" es: 5.
[ luis: ~/Desktop ]$ ls
archivo archivo1 archivo2 archivo3 archivo4 CopiaSeguridad.tar E5_2.sh FS
[ luis: ~/Desktop ]$
```

Vocal en minúscula 'i', directorio existente y archivo 'tar' existente.

```
luis@luis-VirtualBox: ~/Desktop
File Edit View Search Terminal Help
[ luis: ~/Desktop ]$ ./E5_2.sh i ~/Desktop
El argumento tiene 1 sola letra.
El argumento está en minúscula.
El directorio existe.
El archivo "CopiaSeguridad.tar" ya existe.
[ luis: ~/Desktop ]$ ls
archivo archivo1 archivo2 archivo3 archivo4 CopiaSeguridad.tar E5_2.sh FS
[ luis: ~/Desktop ]$
```

Vocal en minúscula 'i', archivo de copia de seguridad no realizada:

```
luis@luis-VirtualBox: ~/Desktop
File Edit View Search Terminal Help
[ luis: ~/Desktop ]$ ls
archivo archivo1 archivo2 archivo3 archivo4 E5_2.sh FS
[ luis: ~/Desktop ]$ ./E5_2.sh i ~/Desktop
El argumento tiene 1 sola letra.
El argumento está en minúscula.
El directorio existe.
Creado archivo CopiaSeguridad.tar
[ luis: ~/Desktop ]$ ls
archivo archivo1 archivo2 archivo3 archivo4 CopiaSeguridad.tar E5_2.sh FS
[ luis: ~/Desktop ]$
```



## 8 Ejemplo de memoria

- **Nombre de los autores:** Luis Baca Ruiz.
- **Grupo:** A2.
- **Titulación:** Doble Grado en Ingeniería Informática y Administración y Dirección de Empresas.
- **Curso:** 1º.

### Enunciado

Crea un Script que muestre por pantalla un argumento que se le pasa por terminal.

### Solución

El siguiente guion contiene las ordenes necesarias para resolver el problema. El mismo script contiene los comentarios que definen qué realiza cada línea.

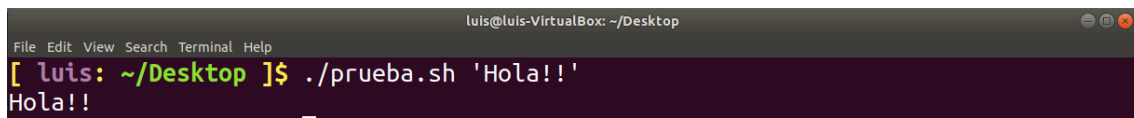
```
#!/bin/bash
# Guardamos el argumento en una variable.
ARGUMENTO=$1

# Mostramos el argumento que se le ha pasado al guion.
echo $1
```

### Pruebas

#### 1

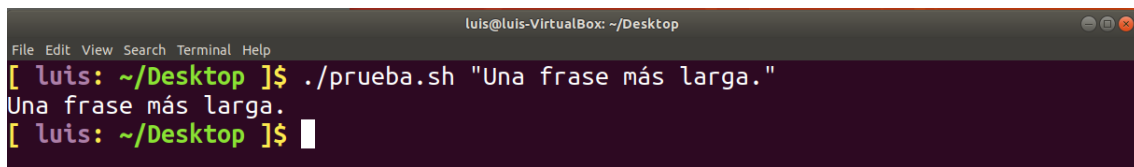
Se ha realizado una prueba pasándole un solo argumento de texto.



```
luis@luis-VirtualBox: ~/Desktop
File Edit View Search Terminal Help
[ luis: ~/Desktop ]$ ./prueba.sh 'Hola!!'
Hola!!
```

#### 2

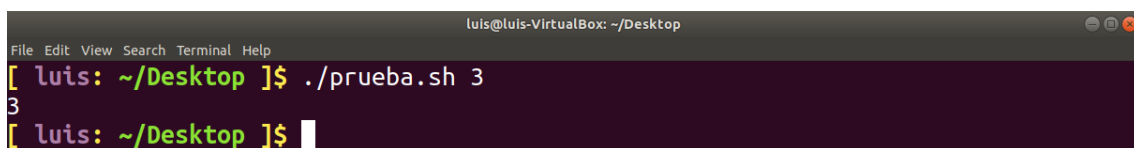
La segunda prueba consiste en pasarle una cadena más larga pero que debe ser tomada como si fuera un solo argumento también.



```
luis@luis-VirtualBox: ~/Desktop
File Edit View Search Terminal Help
[ luis: ~/Desktop ]$ ./prueba.sh "Una frase más larga."
Una frase más larga.
[ luis: ~/Desktop ]$
```

#### 3

En la segunda prueba se le pasa un parámetro numérico esperando que también lo muestre tal y como ha sucedido en los casos anteriores.



```
luis@luis-VirtualBox: ~/Desktop
File Edit View Search Terminal Help
[ luis: ~/Desktop ]$ ./prueba.sh 3
3
[ luis: ~/Desktop ]$
```