Actividad Cooperativa (AC1): **Cómo ejecutar un programa shell**

- Leer individualmente el apartado 2 de los apuntes. 5’
- Repartirse los tres modos de ejecución entre los miembros del grupo.
- Reunirse por grupos de expertos. Hacer las siguientes tareas y aclarar dudas. 10’
- Escribir el siguiente programa llamado *prog1* y ejecutarlo según el modo elegido. Entender lo que ocurre.
- Consultar el valor de la variable *var1* al terminar la ejecución del programa y anotarlo. ¿Por qué tiene ese valor?

```
echo dentro
var1=hola
echo $var1
```

- Escribir el siguiente programa *prog2* y ejecutarlo según el modo elegido. Entender lo que ocurre al ejecutarse la línea del *exit*.

```
echo dentro
var1=hola
echo $var1
exit
var2=adios
echo fuera
echo $var2
```

- Volver al grupo original y explicar a los demás lo aprendido de forma que todos sepan todo. Cada uno tiene 5’ aprox. para explicar su parte a los demás. 15’

Actividad Cooperativa (AC2):

- Repartirse los apartados entre los miembros del grupo.
- Leer individualmente. 5’
- Reunirse por grupos de expertos. Escribir programas. Ejecutarlos. Aclarar dudas. 10’

Asignación de valor y uso de una variable

```
> fruta=pera
> comida=sopa
> echo $fruta $comida
pera sopa
>
```

```
> preposicion=para
> objeto=${preposicion}caidas
> echo $objeto
paracaidas
>
```

```
> s="ls -l"
> $s /usr
obtendremos el listado largo del contenido del directorio /usr
>
```

Utilización de los acentos graves

```
> hoy=`date`
> echo $hoy
Fri Feb 13 10:20:01 GMT 2004
>
```

```
> hoy=date
> $hoy
Obtengo la fecha y hora del momento en que ejecuto el comando
>
```

```
> hoy=date
> echo $hoy
date
>
```

```
> lista=`ls /usr`
> echo $lista
> contenido_fichero=`cat prog1`
> echo $contenido_fichero
```

Asignación externa a variables del shell y variables shell predefinidas

Ejecutar los ejemplos con **read** de los apuntes y el siguiente:

```
echo “Introduce tu nombre”
read nom_usuario
usuario=$LOGNAME
echo “El nombre real del usuario $usuario es $nom_usuario”
```

Introducir ahora datos en la variable del guión que tengan más de una palabra. ¿Qué ocurre?

- Observar el contenido de las variables HOME, PATH y TERM.
- Diferenciar el directorio HOME del actual.
- Modificar el valor de la variable PATH de forma que contenga la ruta al directorio **bin** que cuelga de tu HOME.
- Aprender el uso del comando **cd** sin argumentos.

- Volver al grupo original y explicar a los demás lo aprendido de forma que todos sepan todo. Cada uno tiene 5’ aprox. para explicar su parte a los demás. 15’. El siguiente ejercicio puede servir de repaso.

Escribir el siguiente programa shell:

- Copiar un fichero, cuyo nombre pides al usuario, al directorio que está almacenado en la variable RUTA.
- Guardar en una nueva variable la ruta al directorio actual.
- Mostrar por pantalla el contenido de la variable que has creado y el de la variable HOME.

Actividad Cooperativa (AC3): **Argumentos para los procedimientos shell. shift, $*, $#.**

- Leer todo individualmente. 5'
- Escribir el siguiente programa llamado *mostrar*

```
echo argumento0=$0
echo argumento1=$1
echo argumento2=$2
echo argumento3=$3
```

Ejecutarlo mediante la siguiente línea de comandos:

mostrar Pedro Juan Pepe

Entender lo que ocurre. 10'

- Repartirse: shift, $*, $#
- Reunirse por grupos de expertos. Escribir programas. Entenderlos. Ejecutarlos con 0, 1, 2, 3 y más argumentos. Aclarar dudas. 10'

Opción shift

```
shift
echo argumento0=$0
echo argumento1=$1
echo argumento2=$2
echo argumento3=$3
```

```
shift
shift
echo argumento0=$0
echo argumento1=$1
echo argumento2=$2
echo argumento3=$3
```

```
shift
shift
shift
echo argumento0=$0
echo argumento1=$1
echo argumento2=$2
echo argumento3=$3
```

Opción $*

```
echo $*
echo argumento0=$0
echo argumento1=$1
```

```
echo $*
echo argumento1=$1
echo argumento2=$2
```

```
echo $*
echo argumento1=$1
echo argumento2=$2
echo argumento3=$3
```

Opción $#

```
echo $#
echo argumento1=$1
```

```
echo $#
echo argumento1=$1
echo argumento2=$2
```

```
echo $#
echo argumento1=$1
echo argumento2=$2
echo argumento3=$3
```

- Volver al grupo original y explicar a los demás lo aprendido de forma que todos sepan todo. Cada uno tiene 5' aprox. para explicar su parte a los demás. 15'. Realizar los siguientes ejercicios:

Crear el siguiente guión llamado *argumentos*

```
echo '$0' es $0
echo '$1' es $1
echo '$2' es $2
echo '$*' es $*
echo '$@' es $@
echo '$#' es $#
```

Ejecutar varias veces el guión anterior con uno, dos, tres y más argumentos.
$ *argumentos*
$ *argumentos* uno
$ *argumentos* uno dos
$ *argumentos* uno dos tres

Repetir ahora el ejercicio, pero introduciendo números y cadenas de caracteres:
$ *argumentos* uno
$ *argumentos* uno 2
$ *argumentos* uno dos tres y cuatro
$ *argumentos* uno dos “tres y cuatro”

¿Qué ocurre?, ¿cómo lo explicas?

Crear el siguiente guión:

```
# Nombre del shell: demo_shift
echo “El nombre del programa es $0”
echo “Los argumentos son: $*”
echo “El número de argumentos es $#”
echo “Los tres primeros argumentos son $1 $2 $3”
shift
echo “El nombre del programa es $0”
echo “Los argumentos son: $*”
echo “El número de argumentos es $#”
echo “Los tres primeros argumentos son $1 $2 $3”
shift
shift
echo “El nombre del programa es $0”
echo “Los argumentos son: $*”
echo “El número de argumentos es $#”
echo “Los tres primeros argumentos son $1 $2 $3”
shift 3
echo “El nombre del programa es $0”
echo “Los argumentos son: $*”
echo “El número de argumentos es $#”
echo “Los tres primeros argumentos son $1 $2 $3”
```

Ejecutarlo de la siguiente forma y entender qué ocurre:
demo_shift 1 2 3 4 5 6 7 8 9 10 11 12