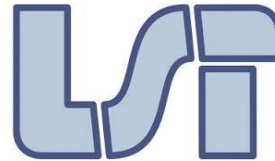


Fundamentos de Software - Prácticas

Módulo I. Órdenes UNIX y Shell Bash

Práctica 4: Expresiones con variables y expresiones regulares

Departamento de Lenguajes y Sistemas
Informáticos – Universidad de Granada



UNIVERSIDAD
DE GRANADA

Práctica 4: Expresiones con variables y expresiones regulares

- OBJETIVOS:

- Definir expresiones con variables mediante operadores aritméticos y relacionales
- Objetivo: poder definir relaciones entre variables
- Evaluación de expresiones aritméticas: `expr`, `$((...))`, `${ ... }`
 - no es necesario dejar huecos en blanco entre los paréntesis más internos y la expresión contenida en ellos
 - las variables no necesitan estar precedidas por el `$`
 - expresiones aritméticas anidadas
- Operadores aritméticos (Tabla 4.2): ojo con `**`, `++`, `--`, `+=`, `-=`, `*=`, `/=`, `%`, `%=`
- `bash` solamente trabaja con números enteros para el resultado del cálculo de expresiones aritméticas: `bc -l`

Práctica 4: Expresiones con variables y expresiones regulares

- OBJETIVOS:

- Definir expresiones con variables mediante operadores aritméticos y relacionales
- Asignar un valor a una variable entera:
 - `y=7`
 - `let variableEntera=expresión`
 - `((expresión))`
 - Uso en expresiones condicionales
- Operadores relacionales:
 - Definir condiciones para expresiones condicionales
 - Su resultado es booleano: 0 (evaluación falsa), 1 (evaluación verdadera) **Ojo: No confundir con \$? (código de retorno de la última orden ejecutada: 0 (correcto), 1 (error))**
 - `==, !=, <, >, <=, >=, !, &&, | |`

Práctica 4: Expresiones con variables y expresiones regulares

- OBJETIVOS:

- Conocer operadores de consulta de archivos y algunas órdenes para utilizarlos.
- Objetivo: Comprobar condiciones relativas a ficheros y directorios dentro de nuestros guiones
 - **Opción 1:** `test -opciones expresión` (Tabla 4.4)
 - 0 (expresión evaluada como true), 1 (evaluación como false)
 - Opciones:
 - Archivo de dispositivo de bloques: los datos se organizan y transmiten en conjuntos indivisibles direccionables en la comunicación con la unidad central de proceso (ej. disco duro)
 - Archivo de dispositivo de caracteres: comunicación con la unidad central de proceso por medio de bytes individuales (ej. Impresoras y teclados).
 - Archivo plano: archivos que se pueden crear, abrir, leer en serie, sobrescribir, truncar y cerrar.
 - Enlace simbólico: indican un acceso a un directorio o un fichero que se encuentra en un lugar distinto dentro de la estructura de directorios.

Práctica 4: Expresiones con variables y expresiones regulares

- OBJETIVOS:

- Conocer operadores de consulta de archivos y algunas órdenes para utilizarlos.
- Objetivo: Comprobar condiciones relativas a ficheros y directorios dentro de nuestros guiones

- **Opción 2:** if/else

```
If condición;  
then  
    declaraciones  
[elif condición;  
    then declaraciones ]...  
[else  
    declaraciones ]  
fi
```

Lista de declaraciones con órdenes (evaluación teniendo en cuenta su estado de finalización: 0 true, 1 false)

Ejecutadas si la condición se evalúa como true

Se accede a los bloques elif si las condiciones previas se han evaluado como false

Se accede al bloque else si todas las condiciones previas se han evaluado como false

Práctica 4: Expresiones con variables y expresiones regulares

- OBJETIVOS:

- Conocer operadores de consulta de archivos y algunas órdenes para utilizarlos.
- Objetivo: Comprobar condiciones relativas a ficheros y directorios dentro de nuestros guiones
 - Opción 2: if/else
 - Ojo: son necesarios los huecos en blanco entre los corchetes y expresión
 - Comparaciones aritméticas: Uso del corchete o del doble paréntesis (ojo con las restricciones usando los dobles paréntesis)
 - `if [$valor == "34"]; if (($var1 <= $var2));`
 - Comparaciones entre cadenas de caracteres: ojo con el espacio en blanco si se utiliza un único =
 - `if [$valor = "hola"]; if [$valor == "adios"]`
 - Ojo con cadenas con espacios en blanco interiores:
 - `if ["$valor" == " hola amigos "];`

Práctica 4: Expresiones con variables y expresiones regulares

- OBJETIVOS:

- Conocer operadores de consulta de archivos y algunas órdenes para utilizarlos.
- Objetivo: Comprobar condiciones relativas a ficheros y directorios dentro de nuestros guiones
 - Opción 2: if/else
 - Comparaciones usando órdenes: ojo: la condición se evalúa a true si la orden se ha ejecutado correctamente y a false en caso contrario
 - `if rm salida; then echo sí; else echo no; fi 2> sal`

Práctica 4: Expresiones con variables y expresiones regulares

- OBJETIVOS:
 - **Conocer el concepto de expresión regular y operadores para expresiones regulares.**
 - Expresión regular: Lenguaje formal para definir patrones que describen un conjunto de cadenas y que se puede utilizar para búsquedas.
 - Conjunto de caracteres (operadores) de la Tabla 4.5
 - Uso con find / grep:
 - `find /etc name "*dev*.conf"`
 - `find /usr/bin -regex '.*\ (cd\ |zip\) .* '`
 - `grep [kyz] /etc/group`
 - Ejemplos: https://bioinf.comav.upv.es/courses/unix/expresiones_regulares.html
 - Probar expresiones regulares: <https://regex101.com/>

Práctica 4: Expresiones con variables y expresiones regulares

- COMANDOS:

- `$((...))`
- `$(...)`
- `bc`
- `let`
- `test`
- `if/else`