**ARRAYS (Vector)**

Los arrays de los script funcionan de la misma forma que los arrays de cualquier lenguaje de programación. Un array es un conjunto o agrupación de valores cuyo acceso se realiza por índices, en un script se puede almacenar en un mismo array todo tipo de cosas, números, cadenas, caracteres, etc.

**Ejemplo:**

|  |  |  |  |
| --- | --- | --- | --- |
| Pedro | Maria | Ana | Luis |

0 1 2 3

En los arrays el primer elemento que se almacena lo hace en la posición 0.

Array de una dimensión, con un único índice numérico, pero sin restricciones de tamaño ni de orden numérico o continuidad.

Los valores de las celdas puedan asignarse de manera individual o compuesta.

El uso de los caracteres especiales [@] o [\*] como índice, supone referirse a todos los valores en su conjunto.

**Declarar un array**

Declare –a nombreArray

**Forma de utilizar**

varArreglo[indice]=valor

A[3]=”Carlos”

Dar valores de golper/ indicando posiciones

A=( valor1 valor2 valor3 [indice]=valorx…. valorN )

**Para referenciar a una celda** o ver el contenido del array en una posición

${varArreglo[idice]}

**Para saber el número de elementos del array**

${ #varArreglo[\*]}

**Ejemplo**

declare –a NUM #declarar el vector

NUM=( cero uno dos tres ) #Asignación compuesta

echo ${NUM[2]} # muestra dos

NUM[4]=”cuatro” #Asignación simple

NUM=( [6]=seis siete [9]=nueve ) #Asignación compuesta en celdas 6 7 y 9

echo ${NUM[7]} #muestra siete

echo ${NUM[\*]} # ? ?

**Ejemplo**

for i in `seq 1 5`

do

A[$i]=$i

done

echo ${A[3]}

**Expresiones de ficheros en condiciones**

-e Fichero Si el fichero existe

-s Fichero Si el fichero no está vacío

-d Fichero Si es directorio

-r Fichero si tiene permiso de lectura

-w Fichero si tiene permiso de escritura

-x Fichero si tiene permiso de ejecución

**Ejemplo**

**#sea un nombre de archivo verifica si existe el fichero si no esta vacio si…**

read archi

if [ -e $archi ]

then

echo “existe el archivo”

fi

if [ -s $archi ]

then

echo “$archi no esta vacio”

fi

if [ -d $archi ]

then

echo “es un directorio ”

else

echo “es un archivo”

fi

if [ -r $archi ]

then

echo “tiene permiso de lectura”

fi

#BUSCA ARCHIVOS O DIRECTORIOS QUE INICIAN EN LA LETRA t EN EL DIRECTORIO DE TRABAJO Y EL DIRECTORIO tcinco

for i in t\* tcinco/t\*

do

if [ -d $i ]

then

echo $i directorio

else

echo $i archivo

fi

done

**EJEMPLOS VARIOS**

**Ejemplo 1.**

#USO DEL FOR

Echo “Introducir inicio y fin del for”

read ini

read fin

for i in `seq $ini $fin`

do

echo $i

done

**Ejemplo 2.**

#uso de parámetros MUESTRA LOS PARAMETROS ENVIADOS

for i in "$@"

do

echo has escrito $i

done

**Ejemplo 3.**

# USO DE VARIABLE PARA EL RECORRIDO DEL FOR

q="uno dos tres cuatro"

for i in $q

do

echo $i

done

**Ejemplo 4.**

#lectura de canciones

read N

for k in `seq 1 $N`

do

read cancion

VC[$k]=$cancion

done

for k in ‘seq 1 $N’

do

echo ${VC[$k]}

done

echo ${VC[\*]}

**Ejercicios propuestos**

1. Almacenar N numero pares en un vector
2. Almacenar N términos de la serie de Fibonacci en un vector
3. Verificar si en un vector existe el elemento x
4. Mostrar la posición donde se encuentra el elemento x
5. Cuantos veces se repite el elemento x
6. Dado el nombre de un archivo verificar si tiene permiso de lectura, escritura y ejecución
7. Dado el nombre de un archivo si no existe crear el archivo en el directorio cuyo contenido sea sus datos
8. Dado el nombre de un archivo si es un directorio crear en ese directorio un archivo datos.txt que contenga sus datos