

Universidad de Chile
Facultad de Ciencias
Departamento de Física

Programación y Métodos Numéricos

Corrección tarea N° 5
Publicada el 31 de Agosto de 2006

Profesor: José Rogan
Ayudantes: María Daniela Cornejo
Max Ramírez
Alejandro Varas

1. La solución mas ingeniosa al problema era usando las variables de entorno. Así, el script queda descrito por las siguientes líneas

```
#!/bin/bash
DESTINO=/home/$USER/$USER-respaldo_$(date +%Y%m%d).tar.bz2
tar -cvjf $DESTINO $HOME
echo ''El respaldo de su cuenta ya esta hecho. Adios''
```

2. La solución al problema es

```
#!/bin/bash
echo "soluciones de la ecuacion cuadratica: a(x^2)+bx+c"
echo -e "ingrese valor de a : \c "
read a
echo -e "ingrese valor de b : \c "
read b
echo -e "ingrese valor de c : \c "
read c
echo $a $b $c > dat1.txt
m='awk '{print 4*$1*$3}' dat1.txt'
n='awk '{print $2*$2}' dat1.txt'
if [ $n -lt $m ]
then
echo la ecuacion no tiene solucion en los reales
elif [ $m -eq $n ]
then
k='awk '{ print (-1*$2)/(2*$1)}' dat1.txt'
echo tiene unica solucion X = $k
eliif [ $m -lt $n ]
then
echo $a $b $c $m $n > dat2.txt
l='awk '{ print ((-1*$2)/(2*$1))- ((sqrt($5 - $4))/(2* $1))}' dat2.txt'
f='awk '{ print ((-1*$2)/(2*$1)) + ((sqrt($5 - $4))/(2*$1))}' dat2.txt'
echo "las soluciones de la ecuacion cuadratica son: X1=$l y X2=$f "
fi
```

3. Para mantener el orden, primero cree dos archivos para mantener el orden. Primero cree el archivo `factorial.bash`, el cual contiene lo siguiente

```
factorial () {
i=$1
k=1
if [ "$1" -lt 0 ]
then
exit
elif [ "$1" -eq 0 ]
then
echo "1"
exit
fi
while [ "$i" -gt 0 ]
do
k=${k*$i}
i=$((i-1))
done
echo "$k"
}
```

y el segundo, donde se define la operación potencia, `potencia.bash`

```
potencia () {
if [ $p = "0" ]
then
echo "El resultado es 1"
else
((aux = n))
for ((i=1;i<=$p;i++))
do
aux=`expr $aux \* $i`
done
echo "$aux"
fi
}
```

Luego, el programa principal `calcula.bash`

```
#!/bin/bash
```

```
source factorial.bash potencia.bash
```

```
echo "Este script pide dos numeros. Si el primero es impar, calculara
su seno mediante la serie de Taylor y cortara esta segun el segundo numero.
Si el primero es par, pero no nulo, calculara el coseno de la misma forma.
```

```

Si es cero, calculara la exponencial de 1, tambien del mismo modo."
echo
echo "Por favor, ingrese dos numeros naturales, el segundo no muy grande"
read A
read B
echo "a=$A, b=$B"
C=echo ${A%2}
if [ "$A" -eq "0" ]
then
i=0
echo "Como a es $A, calcularemos la exponencial de 1 usando la
serie de Taylor."
VALUE=1
while [ $i -le $B ]
do
POWER='potencia ${A+1} ${i+1}'
FACT='factorial $i'
VALUE='echo "$VALUE+$POWER/$FACT" | bc -l'
i=${i+1}
done
echo "La exponencial de 1 es $VALUE"
exit
elif [ "$C" -eq "0" ]
then
i=0
echo "Como a es par no nulo, calcularemos el coseno de a (a=$A)
mediante la serie de Taylor, hasta el termino $B-esimo."
VALUE=0
while [ "$i" -le "$B" ]
do
k=${2*$i}
POWER='potencia $A $k'
FACT='factorial $k'
SIGN='echo "-1^$i" | bc'
VALUE='echo "$VALUE+$SIGN*$POWER/$FACT" | bc -l '
i=${i+1}
done
echo "El coseno de $A a la $B esima potencia es $VALUE"
exit
elif [ "$C" -eq "1" ]
then
i=0
echo "Como a es impar, calcularemos el seno de a (a=$A)
mediante la serie de Taylor, hasta el termino $B-esimo."
VALUE=0
while [ "$i" -le "$B" ]
do
k=${1+2*$i}

```

```
POWER='potencia $A $k'  
FACT='factorial $k'  
SIGN='echo "-1^$i" | bc'  
VALUE='echo "$VALUE+$SIGN*$POWER/$FACT" | bc -l'  
i=$((i+1))  
done  
echo "El seno de $A a la $B esima potencia es $VALUE"  
exit  
fi
```

Nótese lo guiado del `script`, lo cual hace mucho más facil su entendimiento y uso.