**Làm việc với hàm trong Bash shell**

1. Khai báo hàm

Để khai báo hàm trong Bash shell, ta thực hiện như sau:

***function fname()***

***{***

***statements;***

***}***

hoặc

***fname()***

***{***

***statements;***

***}***

với fname là tên của hàm.

2. Gọi hàm

Sau khi hoàn thành việc khai báo hàm, mỗi khi cần sử dụng đến hàm, ta gọi chúng như sau:

Đối với các hàm không có đối số

***$ fname***

Đối với các hàm có đối số, ta thêm danh sách các đối số phía sau tên hàm như sau:

***$ fname arg1 arg2***

với arg1, arg2 là tên của các đối số.

3. Lấy giá trị đối số truyền vào hàm

Sau khi truyền các đối số vào hàm ở bước trên, để có thể sử dụng chúng bên trong hàm, chúng ta sử dụng các biến cục bộ sau:

***$0*** => tên của kịch bản

***$1*** => Đối số thứ 1

***$2*** => Đối số thứ 2

***$n*** => Đối số thứ n

“***$@***” => Danh sách tất cả các đối số được truyền vào.

“***$\****” => In ra danh sách tất cả các đối số truyền vào dưới dạng 1 chuỗi

Ví dụ:

Trong ví dụ này, chúng ta sẽ viết 1 hàm có chức năng cộng 2 số được truyền vào và in ra kết quả.

#!/bin/bash

# Filename: function.sh

function add()

{

num1=$1 --> Lấy giá trị của đối số thứ 1

num2=$2 --> Lấy giá trị của đối số thứ 2

sum=`expr $num1 + $num2` --> Tính tổng 2 số

# In ra kết quả

echo "First number is: " $num1

echo "Second number is: " $num2

echo "$1 + $2 = " $sum

}

Gọi và truyền đối số cho hàm trên

$ add 100 200

Kết quả

First number is: 100

Second number is: 200

100 + 200 = 300

4. Hàm đệ quy

Trong toán học và khoa học máy tính, Đệ quy (recursion) là phương pháp mà trong đó 1 hàm tự gọi đến chính nó. Tức là bên trong thân của hàm có lời gọi đến chính hàm đó. Điều này sẽ tạo ra vòng lặp, vòng lặp này sẽ lặp đi lặp lại cho đến khi thỏa điều kiện dừng, khi đó thì hàm sẽ không gọi đến chính nó nữa.

1 trong những ví dụ rõ ràng của đệ quy là bài toán tìm giai thừa của 1 số

Như chúng ta đã biết:

0! = 1

n! = n x (n-1)!

Có nhiều cách để hiện thực bài toán này trong Bash shell như sử dụng vòng lặp, đệ quy … Trong phần này, chúng ta sẽ sử dụng phương pháp đệ quy để hiện thực bài toán trên.

Khai báo hàm tính đệ quy

function factorial()

{

if [ $1 -eq 0 ]

then

{

echo 1

}

else

{

echo $(($1 \* $(factorial $(($1-1))))) -> Gọi lại hàm factional ở trên

}

fi

}

Chạy hàm trên với n=4

$ factorial 4

24