

Taller de Herramientas Computacionales

Hector Chaparro Reza

Enero 16, 2019

Esta es la octava bitácora del curso intersemestral de Taller de Herramientas Computacionales, con la fecha (16/01/2019). Aquí resumiré lo más relevante de la teoría. Los problemas y las soluciones que surgieron a lo largo de la práctica serán comentadas en el cuerpo de la bitácora.

Comandos bash para un trabajo ms eficiente y bueno problemas, problemas, problemas

1. Bash

(a) Módulos

Con lo aprendido la clase anterior sobre los scrpits surgieron muchos problemas y dudas en la clase sobre como ejecutarlos correctamente ya que a algunos compaeros no les abran los programas. Viendo esto el profesor aclar que tenemos que guardar el mdulo en el que se encontraran nuestros metdos en la misma ubicacin que el script. Tambien aprendimos a ejecutar estos programas de python desde la bash, a usar diferentes ventanas o ms de un idle de python. Tambien jugamos un poco con el shell de python declarando variable y asignando parametros de diferentes formas, uno muy caracterstico de py es el print multilinea. Tambin hicimos conversion de tipos a la hora de imprimir un resultado para que lo vieramos segn nuestras necesidades.

(b) Problemas:

El da anterior tuve problemas con mi computadora, la cual se bloqueo debido a que se apago por falta de pila, lo cual fue todo un relajo ya que no pude codificar los problemas que dejaron de tarea pero hice el analisis, y eso me ayudo a no perder el hilo de la clase. El punto es que el profesor me ayudo a solucionarlo y al principio creamos que fue porque estuve descargando algunos programas.

```
#!/usr/bin/python2.7
# -*- coding: utf-8 -*-
```

```
def mcd(a,b):
    r = a%b
    while r != 0:
        a = b
```

```
b = r
r = a%b
return(b)
```

Lo anterior fue el ejemplo de del problema 1 de como calcular el mximo comn divisor usando el ciclo while, a la hora de estar haciendo mi analisis acertadamente me base en el algoritmo de euclides pero no tena idea de como implementar el ciclo while y una cosa con la que no contaba es que con el signo de porcentaje podemos usar la funcin mdulo de para ahorrarnos trabajo. Tristemente como el profesor dio el ejemplo ahora haba que hacerle actualizacin sin poder usar el signo de porcentaje y esto fue lo que me sali despues de muchos intentos.

```
def macomu(a,b):
r = a-b
while r>b:
a = r
r = a-b
while r != 0:
a = b
b = r
r = a-b
return(b)
```