Actividad 2

Hugo de Jesús Valenzuela Chaparro

25 de febrero de 2016

1. Problema 1

El movimiento se trata de caída libre, es uniformemente acelerado. La aceleración es g y tomaré el eje positivo hacia abajo:

$y = y_0 + v_0 t + \frac{1}{2} g t^2$

pero tomamos $y_0 = 0$ y $v_0 = 0$ por lo que queda

$y = \frac{1}{2} g t^2$

y cuando llega al suelo es la altura h de la torre, entonces:

$h = \frac{1}{2} g t^2 \Rightarrow t = \sqrt{\frac{2h}{g}}$

esa ecuación del tiempo en función de la altura es lo que meteremos al programa.
A continuación el código del programa:

```
from math import sqrt
h = float(input("Proporciona la altura de la torre (metros): "))
t = sqrt((2.0*h)/9.81)
print("El tiempo que tardó la pelota en llegar al suelo es de", t, "segundos")
```

2. Problema 2

Para demostrar eso, deducimos la ecuación de la tercera Ley de Kepler:

$F = G\frac{Mm}{R^2}$; $F = ma$

pero en el movimiento de los planetas la aceleración es la aceleración centrípeta, entonces:

$a = \frac{v^2}{R} \Rightarrow G\frac{Mm}{R^2} = m\frac{v^2}{R}$

tenemos la relación $v = r\omega$ y $\omega = \frac{2\pi}{T}$

$\Rightarrow G\frac{M}{R^2} = \frac{R^2 \frac{4\pi^2}{T^2}}{R}$

resolviendo para el radio R queda:

$R^3 = \frac{GM}{4\pi^2} T^2$

pero las distancias son desde los centros de masa, es decir desde el centro de la Tierra, por eso a R le sumamos el radio R_E de la Tierra y la altura h del satélite.

$\Rightarrow (R_E + h)^3 = \frac{GM}{4\pi^2} T^2$

para pedir el periodo T y que nos de la altura h simplemente despejamos a h de la ecuación

$\Rightarrow h = \sqrt[3]{\frac{GM}{4\pi^2}T^2} - R_E$

Código en python:

```
from math import pi
print ("Este programa le da la altura a la que se encuentra un determinado satelite de la
superficie Terrestre")
T = float(input("Escriba el periodo (segundos) del satelite que orbita la Tierra:"))
h = (((3.983324e+14)*(T*T))/(4*pi*pi))**(1.0/3.0)-6371000
print ("La altura a la que se encuentra el satélite es de", h/1000.0, "km")
```

Un satélite que orbita la Tierra una vez al día tiene una altura de $35860.62510588992km$; uno que lo hace en 90 minutos, tiene una altura de $280.0641802724404km$. A justar por los resultados, no existe un satélite que de una vuelta en 45 minutos pues al meter datos en el programa sale una distancia negativa.

3. Problema 3

Las coordenadas esféricas están dadas por la relación:

$x = \rho \sin\phi \cos\theta$; $y = \rho \sin\phi \sin\theta$; $z = \rho \cos\theta$

Para calcular las coordenadas esféricas a partir de las coordenadas cartesianas tenemos que:

$\rho = \sqrt{x^2 + y^2 + z^2}$

entonces

$\cos\phi = \frac{z}{\rho} \Rightarrow \phi = \cos^{-1}\frac{z}{\rho}$

y

$\cos\theta = \frac{x}{\rho\sin\phi} \Rightarrow \theta = \cos^{-1}\frac{x}{\rho\sin\phi}$

A continuación el código del programa:

```
from math import sin,acos,pi,sqrt
x = float(input("Escribe la coordenada x"))
y = float(input("Escribe la coordenada y:"))
z = float(input("Escribe la coordenada z:"))
r = sqrt(x*x+y*y+z*z)
phi = acos(z/r)
theta = acos(x/(r*sin(phi)))
print("r =",r,"phi =",phi*180/pi, "grados", "theta =", theta*180/pi, "grados")
```

4. Programa 4

Este programa que pide un número entero impar y otro número entero par, funciona con un condicional que es ”while”. Lo que hace es usar la función intrínseca de Python llamada residuo y representada por ” %”, dicho residuo es lo que sobra al dividir un número entre dos. El condicional vuelve a pedir los números al usuario, si éste no los ingreso bien, siempre que el residuo sea cero. Esto funciona porque si sumamos dos núneros enteros o dos números pares dará como resultado un número par, por lo que el residuo será igual a 0, pero si sumamos uno par y otro impar dará como resultado un número impar y el residuo será distinto de 0.

A continuación el código en Python:

```
print("Enter two integers, one even, one odd.")
m = int(input("Enter the first integer: "))
n = int(input("Enter the second integer: "))
while (m+n)%2==0:
    print("One must be even and the other odd.")
    m = int(input("Enter the first integer: "))
    n = int(input("Enter the second integer: "))
print("The numbers you chose are",m,"and",n)
```

5. Programa 5

Para calcular los números de Catalán utilicé la formula $\frac{(2n)!}{(n+1)!n!}$,con $n \geq 0$. En el código usé un condicional, para calcular todos los números de Catalán siempre y cuando fueran menor o igual que un millón.

A continuación el código en Python:

```
from math import factorial
cn=0
n=0
while cn <= 1000000:
    cn = (factorial(2*n))/(factorial(n+1)*factorial(n))
    if cn <= 1000000:
        print(cn)
    n += 1
```