**PRÁCTICA PARTE I: FUNCIONES MATEMATICAS EN LENGUAJE PYTHON**

**PRÁCTICA PARTE I: FUNCIONES MATEMATICAS**

**FUNCION I:**

import math

def Funcion1(x):

"""Devuelve un valor f(x), siempre y cuando el valor dado sea mayor que -1"""

if x > -1:

x = (x + ((abs(x))\*\*0.25))\*\*0.5\*\*-x

return x

else:

print "El valor x debe de ser mayor que -1"

**FUNCION II:**

"""Devuelve un valor f(x), siempre y cuando el valor dado sea diferente que

0"""

**def Funcion2(x):**

**if x != 0:**

**x = ((math.e)\*\*((x - 1)/math.sin(x)))\*\*0.333333333**

**return x**

**else:**

**print "El valor x debe de ser diferente de 0"**

**FUNCION III:**

"""Devuelve un valor f(x) entre el intervalo

de todos lo números reales"""

f = ((2)\*\*x\*\*2) + ((3)\*\*y\*\*3)

return f

**FUNCION IV:**

def Funcion4(x):

"""Devuele un valor f(x) siempre y cuando x sea mayor o igual que 0 """

if x >= 0:

x = (x\*\*(math.e + 1))/((abs(x - 2) + 3)\*\*0.33333333333333)

return x

else

print "El valor de X debe ser mayor o igual que 0"

**FUNCION V:**

def Funcion5(x,y):

"""Devuelve una valor f(x,y) siempre y cuando

x>0 & x^(y+1) !=0 """

if x > 0 and (x\*\*(y+1))!= 0:

f = (x + 1)\*((x + (y /(x\*\*(y + 4))))/((x\*\*(2 - y) + (1/(x\*\*(y-2))))))

return f

else:

print "El valor de x tiene que cumplir las siguientes condiciones:x>0 & x^(y+1) !=0"

**FUNCION VI:**

def Funcion6(x,y):

"""Devuele un valor f(x) entre el intervalo

de todos lo numeros reales"""

f = abs(math.cos(x)+math.sin(y)) \* (x \*\* 0.34654)

return f

**FUNCION VII:**

def Funcion7(x,y):

"""Devuelve una valor f(x,y) siempre y cuando

x este en el intervalo de -1 y 1"""

if x < 1 and x > -1:

f = (5 \* math.asin(x)) - ( x - (y/2))

return f

else:

print "El valor x debe de estar en un rango de -1 a 1"

**FUNCION VIII:**

def Funcion8(x,y,z):

"""Devuelve una valor f(x,y) siempre y cuando

Z este en el intervalo de -1 y 1"""

if z < 1 and z > -1:

f = ((math.asin(z))\*\*2) + abs(x + y)

return f

else:

print "El valor z debe de estar en un rango de -1 a 1"

**FUNCION IX:**

def Funcion9(u,v):

"""Devuelve una valor f(x,y) siempre y cuando

V este en el intervalo de -1 y 1"""

if v < 1 and v > -1:

f = ((math.e)\*\*(u-1))+ math.acos(v)

return f

else:

print "El valor v debe de estar en un rango de -1 a 1"

**FUNCION X:**

def Funcion10(u,v):

"""Devuelve una valor f(x,y) siempre y cuando

V sea diferente de 0"""

if v != 0:

f = ((abs(u))\*\*0.5) + ((v\*\*3)/((v\*\*3)/(u + (v\*\*3))))

return f

else:

print "El valor v debe ser diferente de 0"

**FUNCION XI:**

def Funcion11(x,y):

"""Devuelve un valor f(x)con soluicion en todos los reales"""

f = ((x + ((abs(y))\*\*0.25))\*\*0.33333333) + math.log(x[8])

return f

**FUNCION XII:**

def Funcion12(x,y):

"""Devuelve un valor f(x) con solucion en todos los reales"""

f = (1 - (math.tan(math.sqrt(x)))\*\*2) + (1/((math.cos(math.sqrt(x)))\*\*2))

return f

**PRÁCTICA PARTE II: FUNCIONES DISCONTINUAS EN PYTHON**

**(parte de Luis aquí cuando termine)**

**PRÁCTICA PARTE III: FUNCIONES DISCONTINUAS**

**(parte de angel aquí cuando termie)**

**PRÁCTICA PARTE IV: METODO DE ELIMINACION DE GAUSS JORDAN**

**from \_\_future\_\_ import division**

**def crearMatriz(p, q):**

**return [[0 for j in range(q)]for i in range (p)]**

**def entrarDatos(matriz, p, q):**

**for i in range(p):**

**for j in range(q):**

**matriz[i][j] = float(input("? "))**

**return**

**def mostrarDatos(matriz, p, q):**

**for i in range(p):**

**for j in range(q):**

**print matriz[i][j],"\t",**

**print "\n"**

**return**

**"""Lo primero seria colocar el sistemas de ecuaciones en notacion matricial**

**de forma que queden solo los coeficienfes de cada variable; ejemplo '2x+3y-z=5'**

**es equivalente a '2, 3, -1, 5' """**

**"""Aqui tenemos el sistema de tres ecuaciones puesto en notacion matricial"""**

**A = [[2,-1,1,2],[3,1,-2,9],[-1,2,5,-5]]**

**def gauss(A, m, n):**

**"""Aplica el metodo de eliminacion de Gauss a una matriz 'A', de m filas**

**y n columnas. Tras la transformacion, la matriz queda en forma triangular**

**donde los elementos debajo de la diagonal principal van a ser cero."""**

**for i in range(n):**

**for k in range(i + 1, n):**

**p = -A[k][i] / A[i][i]**

**for j in range(i, n + 1):**

**if (i == j):**

**A[k][j] = 0**

**else:**

**A[k][j] = A[k][j] + p \* A[i][j]**

**""" Se utilizaron las siguientes expresiones para hacer los elementos de la**

**diagonal superior cero: 1)f2-> f2 - 2\*f1**

**2)f3 ->f3 - 3\*f1"""**

**for i in range(0,n):**

**print A[i][n]**

**def transposicion(A, m, n):**

**T = crearMatriz(m,n)**

**for i in range(n):**

**for j in range(m):**

**T[j][i] = A[i][j]**

**return T**

**def gaussJordan(A, m, n):**

**"""Aplica el metodo de eliminacion de Gauss-Jordan a una matriz A, de m filas**

**y n columnas. La matriz 'A' debe encontrarse YA EN FORMA TRIANGULAR. Tras la**

**transformacion, la matriz queda en forma reducida ."""**

**for i in range(n - 1, -1, -1):**

**for k in range(i - 1, -1, -1):**

**if (A[i][i] == 0):**

**p = A[i][i]**

**else:**

**p = -A[k][i] / A[i][i]**

**for j in range(i, -2, -1):**

**if (i == j):**

**A[k][j] = 0**

**else:**

**A[k][j] = A[k][j] + p \* A[i][j]**

**for i in range(m):**

**for j in range(n):**

**if (A[i][j] != 0 and A[i][j] != 1):**

**t = A[i][j]**

**for k in range(0, n + 1):**

**A[i][k] = A[i][k] / t**

**for i in range(0,n):**

**print "Solucion:", A[i][n]**

**def main():**

**print "La matriz que se ha ingresado es:"**

**mostrarDatos(A,3,3)**

**gauss(A,3,3)**

**print "Las soluciones del sistema de ecuaciones X,Y,Z respectivamente son:"**

**gaussJordan(A,3,3)**

**transposicion(A, 3, 3)**

**main()**