Códigos del tema Cálculo de raíces.
Curso de Física Computacional

M. en C. Gustavo Contreras Mayén

1. Incrementos sucesivos

```
def buscaraiz(f,a,b,dx):
    x1 = a; f1 = f(a)
    x2 = a + dx; f2 = f(x2)
    while f1*f2 > 0.0:
        if x1 >= b: return None
        x1 = x2; f1 = f2
        x2 = x1 + dx; f2 = f(x2)
    else:
        return x1,x2
```

2. Bisección

```
    def bisect(f,x1,x2,switch,epsilon=1.0e-9):
    f1 = f(x1)
    if f1 == 0.0: return x1
    f2 = f(x2)
    if f2 == 0.0: return x2

    if f1*f2 > 0.0: print 'La raiz no se ha identificado en un intervalo'

    n = ceil(log(abs(x2 - x1)/epsilon)/log(2.0))

    for i in np.arange(n):
        x3 = 0.5*(x1 + x2); f3 = f(x3)
        if (switch == 0) and (abs(f3) > abs(f1)) \
                         and (abs(f3) > abs(f2)):
            return None
        if f3 == 0.0: return x3
        if f2*f3 < 0.0:
            x1 = x3; f1 = f3
        else:
            x2 =x3; f2 = f3
    return (x1 + x2)/2.0
```

2.1. Estrategia de solución

```
def f(x): return x**3-10*x**2+5

a,b,dx = (-2.0,11.0,0.02)

print 'Intervalo (x1,x2)    raiz'
while 1:
    try:
        x1, x2 = buscaraiz(f,a,b,dx)
    except Exception, e:
        print e; break
    if x1 != None:
        a = x2
        root = bisect(f,x1,x2,0)
        if raiz != None: print '(%2.4f, %2.4f) %2.8f' %(x1, x2, raiz)
    else:
        print '\nHecho'
        break
```

3. Newton-Raphson

```
def newtonRaphson(f,df,a,b,tol=1.0e-9):
    fa = (f(a)
    if fa == 0.0: return a
    fb = f(b)
    if f(b) == 0.0: return b
    if fa*fb > 0.0: print 'La raiz no esta en el intervalo'
    x = 0.5 * (a + b)

    for i in range(30):
        fx = f(x)
        if abs(fx) < tol: return x

        if fa*fx < 0.0:
            b = x
        else:
            a = x; fa = fx
        dfx = df(x)

        try: dx = -fx/dfx
        except ZeroDivisionError: dx = b - a
        x =  x + dx

        if(b - x)*(x - a) < 0.0:
            dx = 0.5*(b-a)
            x = a + dx

        if abs(dx) < tol*max(abs(b),1.0): return x

    print 'Son demasiadas iteraciones'
```