Instituto Tecnológico de Ensenada

Métodos Numéricos

Lic. Romeo Antonio Gómez Espinoza Examen 2: Raíces de ecuaciones.

26 de septiembre de 2019

1. [40 puntos] Describir geométricamente, ventajas y desventajas de los métodos de bisección, punto fijo, Newton-Raphson y secante.

2. [10 puntos] Escribir el pseudocódigo del método de Newton-Raphson.

3. [10 puntos] Escribir el pseudocódigo del método de la secante.

4. [40 puntos] Sea $f(x) = x^5+3x^4-5x^3-15x^2+4x+12$. Encontrar las 5 raíces
(con cada método) utilizando los métodos de bisección, punto fijo, Newton-Raphson y secante y escribir las iteraciones que tarda en llegar a la raíz con 11 cifras significativas. En el método de bisección especificar los extremos del intervalo y en los demás especificar las aproximaciones iniciales que utilizaron.

ZEMPOALTECATL MEJIA ITSAI
16760300

1.-

Método	Ventajas	Desventajas
Bisección	-Es siempre convergente - Es óptimo para resolver una ecuación $f(x)=0$ cuando no se sabe nada de f, excepto calcular su signo. -Requiere que f sea continua en el intervalo especificado. -Se puede establecer el límite de error.	-No tiene en cuenta a magnitud de los valores de la función en las aproximaciones calculadas, sólo tiene en cuenta el sigo de $f(x_n)$, por lo que hace que buenas aproximaciones intermedias pasen desapercibidas. -Su convergencia es muy lenta -No puede determinar raíces complejas
Punto fijo	-Converge con rapidez -Cuando converge es de mucha precisión. -No necesita de un intervalo para funcionar sino de únicamente un punto perteneciente al intervalo donde esté la raíz. -Posee condiciones para asegurar la convergencia	-La convergencia depende de la magnitud de g(x) -Necesidad de construir funciones g(x) para iterar. -La función correcta de g(x) puede ser muy compleja de encontrar. -Hay infinidad de g(x) y no existe una regla para escoger la correcta.
Newton Raphson	-Convergencia rápida. -El método converge cuadráticamente para raíces simples y linealmente para raíces multiples. -Encuentra raíces complejas.	-Necesita calcular derivada. -No se puede prever la cantidad de iteraciones a partir de una cota de error. -No siempre converge.
Secante	-Se puede aplicar cuando la función f(x) es demasiado compleja como para obtener su derivada. -Procede independientemente de los signos de la función, no se tiene en cuenta el signo de la función para estimar el siguiente punto. -Es un proceso iterativo y por lo mismo , encuentra la aproximación con casi la misma rapidez que el método de Newton-Raphson	-Al ser un proceso iterativo corre el mismo riesgo que el método de newton de no converger a la raíz.

2.-

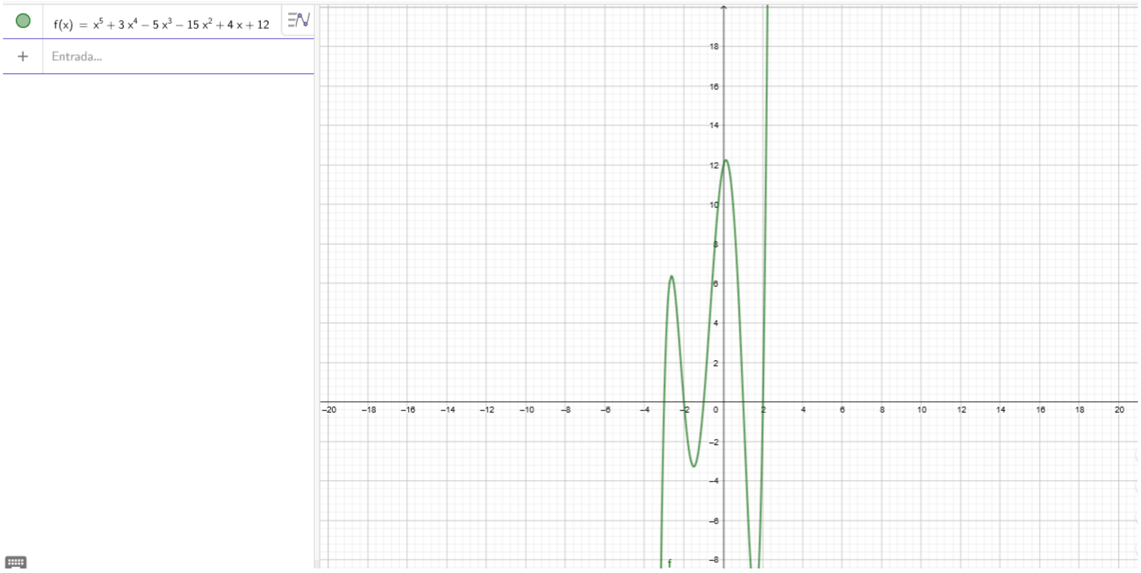
```
#Metodo de Newton-Raphson
i=1

while i <= n:


    p= p0 - (f(p0) / fp(p0))

    if abs(p - p0) < e:
        print(p)
        break
    end if

    p0=p
    i+=1
end while
```

3.-

```
#Metodo de la secante
i=2

q0=f(p0)
q1=f(p1)

while i <= n:

    p= p0 - (q0 * (p1 - p0) / (q1 - q0))
    if abs(p-p1) < e:
        print(p)
        break
    end if

    p0 = p1
    p1 = p
    q0 = q1
    q1 = f(p)

    i+=1
end while
```

ZEMPOALTECATL MEJIA ITSAI
16760300

4.-



Por bisección (parámetro a= -3, b= -1)

```
C:\Users\itsai\Desktop\UNIVERSIDAD\MET_NUMERICOS>python biseccion.py -A -3 -B -1 -F x**5+3*x**4-5*x**3-15*x**2+4*x+12
 -E 0.00000000001 -N 100

It: 1 Solucion a la ecuación x**5+3*x**4-5*x**3-15*x**2+4*x+12 = -2.0, fc=0, fa=0
```

Por punto fijo (aproximación inicial p0=1)

```
C:\Users\itsai\Desktop\UNIVERSIDAD\MET_NUMERICOS>python puntofijo.py -F x**5+3*x**4-5*x**3-15*x**2+4*x+12 -G x**5+3*x
**4-5*x**3-15*x**2+5*x+12 -P 1 -E 0.00000000001 -N 100
It: 1 - p= 1.00000000000000 - f(p0)= 0
```

Por Newton-Raphson (aproximación inicial p0= 0.6)

```
C:\Users\itsai\Desktop\UNIVERSIDAD\MET_NUMERICOS>python newton_raphson.py -I 0.6 -F x**5+3*x**4-5*x**3-15*x**2+4*x+12
 -D 5*x**4+12*x**3-15*x**2-30*x+4 -E 0.00000000001 -N 100
It: 1 - p= 1.11897029702970, f(p0)= 8.38656000000000, f'(p0)= -16.1600000000000
It: 2 - p= 0.998571762959363, f(p0)= -2.85334223131988, f'(p0)= -23.6991442906845
It: 3 - p= 1.00000017415379, f(p0)= 0.0342735597585917, f'(p0)= -23.9941831121324
It: 4 - p= 1.00000000000000, f(p0)= -0.00000417969111499161, f'(p0)= -24.0000006966136
It: 5 Valor de p= 1.00000000000000, f(p0)= -5.77315972805081E-14, f'(p0)= -24.0000000000000
```

Por Secante (p0 = 0, p1 = 2)

```
C:\Users\itsai\Desktop\UNIVERSIDAD\MET_NUMERICOS>python secante.py -A 0 -B 2 -F x**5+3*x**4-5*x**3-15*x**2+4*x+12 -E
0.00000000001 -N 100
It: 2 Valor de p= 2.00000000000000, q0= 12, q1= 0
```

ZEMPOALTECATL MEJIA ITSAI
16760300