

METODOS NUMERICOS

**UNIDAD 6 ACTIVIDADES INTEGRADORAS**

CONEJO EROSA JESUS GUSTAVO

UNIDAD 1 CODIGO EN JAVA METODO ITERATIVO

METODO DE INTERVALO CODIGO

Entradas: f, a, b, Nmax, tol

fa = f(a)

if fa = 0

"a es raíz aproximada"

stop

end if

fb = f(b)

if fb = 0

"b es raiz aproximada"

stop

end if

if signo(fa) = signo (fb)

stop

end if

error = b-a

c = (a+b)/2

fc = f(c)

k = 1

while k <= Nmax and error > tol y fc != 0

aux = c

if signo(fc) != signo (fa)

b = c

fb = fc

else

a = c

fa = fc

end if

c = (a + b) / 2

fc = f(c)

error = |c-aux|

k = k + 1

end while

if fc = 0

"raiz aproximada x = c"

else if error <= tol

"raiz aproximada x=c con error = error"

else

"superado #maximo de iteraciones"

end

stop

**METODO DE APROXIMACION SUCESIVAS**

public abstract class Ecuacion {

protected static final double ERROR=0.001;

public double raiz(double x0){

double x1;

while(true){

x1=f(x0);

if(Math.abs(x1-x0)<ERROR) break;

x0=x1;

}

return x0;

}

abstract public double f(double x);

}

Las clases derivadas denominadas Funcion1 y Funcion2 definen la función f(x)

public class Funcion1 extends Ecuacion{

public double f(double x){

return Math.cos(x);

}

}

public class Funcion2 extends Ecuacion{

public double f(double x){

return Math.pow(x+1, 1.0/3);

}

}

Creamos objetos de las clases derivadas y llamamos desde ellos a la función raiz que describe el procedimiento numérico

Funcion1 f1=new Funcion1();

System.out.println("solucion1 "+f1.raiz(0.5));

System.out.println("solucion1 "+f1.raiz(0.9));

Funcion2 f2=new Funcion2();

System.out.println("solucion1 "+f2.raiz(0.5));

**METODO DE INTERPOLACION**

#include <cstdlib>

#include <iostream>

using namespace std;

int main(int argc, char \*argv[])

{

float x[50],y[50],z,l, valor=0;

int n;

cout<<"ingrese el numero de elementos: "; cin>>n;

cout<<"ingrese x: ";

for(int i=0; i<n; i++){

cin>>x[i];

}

cout<<endl;

cout<<"ingrese y: ";

for(int i=0; i<n; i++){

cin>>y[i];

}

cout<<endl;

cout<<endl<<"ingrese z: "; cin>>z;

for(int i=0; i<n ;i++){

l=y[i];

for(int j=0; j<n; j++){

if(i!=j){

l=(l\*(z-x[j]))/(x[i]-x[j]);

}

}

valor=valor+l;

}

cout<<endl<<endl<<"El valor al polinomio de interpolacion en Z = "<<z<<" es : "<< valor<<endl;

system("PAUSE");

return EXIT\_SUCCESS;}