



**Instituto Tecnológico de Ciudad Guzmán**

**Ingeniería en Sistemas Computacionales**

**Departamento de Sistemas e Informática**

**Métodos Numéricos**

**Reporte del programa de métodos iterativos con aproximación de error**

**Chávez Jiménez Luis Fernando, 19290903**

**Moreno Rivera Guillermo, 19290933**

**Prof. Alejandro Arturo Rivera Sánchez**

## Función Seno

```
//Calculo seno de x
public static double seno(double n){
    double eA = 10;
    double temp = 0;
    double tempSuma = 0;
    double tempSumaA = 0;
    int c = 0;
    while(eA > errorMeta){
        temp = (pow(-1, c) / factorial((2*c)+1) * pow(n, (2*c) + 1));
        if(c == 0){
            tempSuma = temp;
        } else {
            tempSumaA = tempSuma;
            tempSuma = tempSuma + temp;
            eA = calEA(tempSuma, tempSumaA);
        }
        c++;
    }
    return tempSuma;
}
```

```
Ingrese un valor x para realizar el cálculo: 2
El valor de x se encuentra en:
1. x en radianes
2. x en grados
Ingrese su respuesta: 1
Su resultado es: 0.909347442680776
```

## Función Coseno

```
//Cálculo de coseno
public static double coseno(double x){
    double sumando, sumatoria = 0;
    int n = 0;
    do{
        sumando = pow(-1, n) / factorial(2 * n) * pow(x, 2*n);
        sumatoria = sumatoria + sumando;
        n = n + 1;
    }while (absD(sumando) > errorMeta);
    return sumatoria;
}
```

Ingrese un valor x para realizar el cálculo: 2  
El valor de x se encuentra en:  
1. x en radianes  
2. x en grados  
Ingrese su respuesta: 1  
Su resultado es: -0.4222222222222223

## Función Tangente

```
//Cálculo de tangente de x
public static double tangente(double n){
    return seno(n) / coseno(n);
}
```

Ingrese su valor x para hacer el cálculo: 2  
El valor de x se encuentra en:  
1. x en radianes  
2. x en grados  
Ingrese su respuesta: 1  
Su resultado es: -2.153717627401838

## Raíz Cuadrada

```
//Calculo de la raíz cuadrada
public static double raiz(double x){
    double b = x, errorAct, estAnt = 1;
    errorAct = 10000;
    while (errorAct > errorMeta){
        b = ((x/b) + b) / 2;
        errorAct = calEA(estAnt, b);
        estAnt = b;
    }
    return b;
}
```

Ingrese un valor x para realizar el cálculo: 2  
Su resultado es: 1.4142135623746899

## Logaritmo Natural

```
//Calculo de Ln
public static double logaritmoN(double n){
    errorMeta = errorMeta * 0.01;
    double eA = 10;
    double temp = 0;
    double tempSuma = 0;
    double tempSumaA = 0;
    double j = 0;
    int c = 0;
    while (eA > errorMeta) {
        //while (c!= 5) {

        j = (2*c)+1;
        temp = 2 * ((1/(j))*(pow(((n-1)/(n+1)), j)));

        if(c == 0){
            tempSuma = temp;
        } else {
            tempSumaA = tempSuma;
            tempSuma = tempSuma + temp;
            eA = calEA(tempSuma, tempSumaA);
        }
        c++;
    }
    return tempSuma;
}
```

Ingrese una opción: 4  
Ingrese su valor x para hacer el cálculo: 10  
Su resultado es: 2.3024452417230443

## Exponencial

```
//Exponencial
public static double exp(double x){
    double sumando, sumatoria = 0, errorAct;
    errorAct = 10000;
    int n = 0;
    while (errorAct > errorMeta){
        if(n == 0) sumando = 1;
        else if(n == 1) sumando = x;
        else{
            sumando = pow(x, n)/factorial(n);
        }
        sumatoria = sumatoria + sumando;
        n = n + 1;
        errorAct = calEA(sumatoria, (sumatoria-sumando));
    }
    return sumatoria;
}
```

```
//Exponencial
public static double exp(double x){
    double sumando, sumatoria = 0, errorAct;
    errorAct = 10000;
    int n = 0;
    while (errorAct > errorMeta){
        if(n == 0) sumando = 1;
        else if(n == 1) sumando = x;
        else{
            sumando = pow(x, n)/factorial(n);
        }
        sumatoria = sumatoria + sumando;
        n = n + 1;
        errorAct = calEA(sumatoria, (sumatoria-sumando));
    }
    return sumatoria;
}
```