

Software sobre el método de Newton Raphson

Armando Pineda Paredes

Código: 191697

Universidad Francisco De Paula Santander Ocaña

Facultad De Ingeniería

Ingeniería De Sistemas

Análisis numérico

Norte De Santander

Ocaña

27/04/2021

Ocaña, N.S

Software sobre el método de Newton Raphson

Armando Pineda Paredes

Código :191697

Docente:

Álvarez Quintero Orlando

Universidad Francisco De Paula Santander Ocaña

Facultad De Ingeniería

Ingeniería De Sistemas

Análisis numérico

Norte De Santander

Ocaña

27/04/2021

Ocaña, N.S

INDICE

| | |
|------------------------------------------------------------------------------|-----------|
| SOFTWARE SOBRE EL METODO DE NEWTON RAPHSON | 5 |
| Objetivos del método de newton Raphson..... | 6 |
| Objetivo Principal..... | 6 |
| Objetivos específicos..... | 6 |
| Justificación Sobre El Método De Newton-Raphson | 7 |
| Marco Teórico Del Método De Newton Raphson..... | 9 |
| Marco Teórico Del Método De Newton Raphson..... | 12 |
| ¿Cuáles son las características de Python? | 12 |
| ¿Qué se puede hacer con Python?..... | 13 |
| Metodología | 14 |
| Resultados..... | 16 |
| EJEMPLO 1..... | 17 |
| EJEMPLO 2..... | 19 |
| Diagrama de flujo del software sobre el método de newton Raphson..... | 22 |
| Código Fuente..... | 23 |
| Guía de usuario | 48 |
| 1. Menú del software..... | 48 |
| 1.1. Menú del software – Archivo:.. | 49 |
| 1.2. Menú del software – Editar | 51 |
| 1.3. Menú del software – Guía usuarios..... | 51 |
| 1.4. Menú del software – Ayudas | 54 |
| 1.5. Menú del software – Contacto | 55 |
| 2. Cajas de Texto, para ingresar los datos | 58 |
| 2.1. Caja de texto – Función $F(X)$:..... | 58 |
| 2.2. Caja de texto – Derivada de $F(X)$: | 59 |
| 2.3. Caja de texto – Valor inicial: | 60 |
| 2.4. Caja de texto – Error Relativo: | 60 |
| 3. Botones..... | 61 |
| 3.1. Botón – Resolver:..... | 61 |
| 3.2. Botón – Limpiar tabla: | 62 |
| 4. Widgets del software..... | 62 |

| | |
|------------------------------------------|----|
| 4.1 Widgets – Widgets de pregunta: | 62 |
| 4.2 Widgets – Widgets de reloj:..... | 63 |
| Conclusiones | 64 |
| Bibliografía | 65 |
| METODO DE NEWTON RAPSHON | 65 |
| LENGUAJE DE PROGRAMACION PYTHON | 65 |
| INTERFAZ GRAFICA TKINTER | 65 |
| LOGICA PYTHON..... | 65 |

SOFTWARE SOBRE EL METODO DE NEWTON RAPHSON

El método de newton-Raphson es llamado así por el matemático inglés Joseph Raphson (contemporáneo de newton) quien en su libro *Aequationum Universalis*, publicado en 1660, contenía este método para aproximar raíces, newton en su libro *Método de las fluxiones* describe el mismo método en 1671 pero no fue publicado hasta 1736

El método numérico de newton es una aplicación del cálculo diferencial y numérico que se utiliza para hallar los ceros de una función derivable de enésimo grado con la precisión deseada ya que es una extensión directa del método del mismo nombre para buscar ceros de funciones de una variable, los procedimientos para hallar las raíces o ceros de funciones lineales o cuadráticas a partir de los coeficientes de la ecuación son sencillos y exactos

El método de newton asume que la función F sea continuamente derivable y que se conoce la derivada de la función, este método puede no converger si se comienza con un valor muy alejado de la raíz, sin embargo, si converge lo hace mucho más rápido que el método de bisección, el método de newton también es útil por que se generaliza para problemas de dimensiones más altas.

El método de newton Raphson parte de un valor inicial que se introduce en una expresión relacionada con la ecuación, obteniendo así un resultado, ese resultado se introduce en la misma expresión, obteniendo un nuevo resultado y así sucesivamente, si la elección del valor inicial es

buena, cada vez que introducimos unos de los resultados obtenidos en esa expresión (es decir, cada vez que realizamos una iteración del método) el método nos proporciona una aproximación a la solución real mejor que la que tuviéramos anteriormente

Objetivos del método de newton Raphson

Objetivo Principal

El objetivo de este método es para estimar la solución de una ecuación $F(X)=0$ es producir una sucesión de aproximación que se acerquen a la solución (iteraciones o error relativo).

Escogemos el primer número x_0 de la secuencia y luego en circunstancias favorables el método hace uso de la derivada de la función para estimar un posible valor de la raíz, partiendo de uno inicial

Objetivos específicos

- Dada una función $f(x)=0$ obtener la primera y segunda derivada
- Elegir un valor inicial X_0 , este valor inicial debe cumplir con el criterio de convergencia
- Obtener una nueva aproximación evaluando la formula general del método

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)},$$

- Encontrar el error relativo ingresado por el usuario

Justificación Sobre El Método De Newton-Raphson

La presente indagación tiene el objetivo de contribuir al establecimiento del entendimiento científico sobre la educación de la Matemática por un lado y el aprendizaje por el otro, por medio de la aplicación la predominación de la aplicación del Software matemático del método de newton Raphson en Python como táctica didáctica en la resolución del proceso educativo de la materia análisis numérico.

Por otro lado, se vio en nuestra realidad que se carece de nuevas tácticas de educación y aprendizaje de la matemática por parte de la educación que todavía no ha conseguido integrarse de forma eficiente en hablado proceso, significativo se use el recurso educativo conveniente como una forma de tecnología en especial por las estudiantes y profesores.

producto de las más sofisticadas tecnologías o lenguajes de programación en los cuales siempre se ha encontrado Python, la enseñanza no puede estar al margen de esta innovación, más por otro lado los maestros y alumnos tienen que estar a la vanguardia en el dominio y la implementación de este instrumento tecnológica.

Al final el presente proyecto se hace con el objetivo de brindar tanto al estudiante como al maestro accesibilidad a software matemáticos ayudando a la eficiencia del aprendizaje, usando técnicas que tengan directa interacción con las habilidades y el aprendizaje de los alumnos que tienen que complementarse.

Enlazando unos con otros para Por las causas expuestas, nace la necesidad de hacer este trabajo de desarrollo de software que expone la implementación por medio de la aplicación de la predominación del Programa matemático de newton Raphson como táctica didáctica, que posibilita mejorar el grado resolución de inconvenientes de ecuaciones no lineales de una variable en Renovables de la universidad francisco de paula Santander Ocaña, realizando que el proceso de educación – aprendizaje sea realmente relevante y conseguir los objetivos esperados

Marco Teórico Del Método De Newton Raphson

En el análisis numérico, el método de Newton (también conocido como el método de Newton-Raphson), llamado así por Isaac Newton y Joseph Raphson, es un método para encontrar sucesivamente mejores aproximaciones a las raíces (o ceros) de una función de valor real.

Requisitos previos del método

¡Se debe partir de una función $f(x)$ que sea al menos una vez derivable, tenga una primera derivada continua con el mismo signo en ese rango y que $f'(x) \neq 0$. Se pueden encontrar casos de lenta convergencia cuando se encuentra algún punto de inflexión [$f''(x)=0$] en la vecindad de una raíz, el método oscila alrededor de un mínimo o máximo local o se encuentran pendientes cercanas a cero.

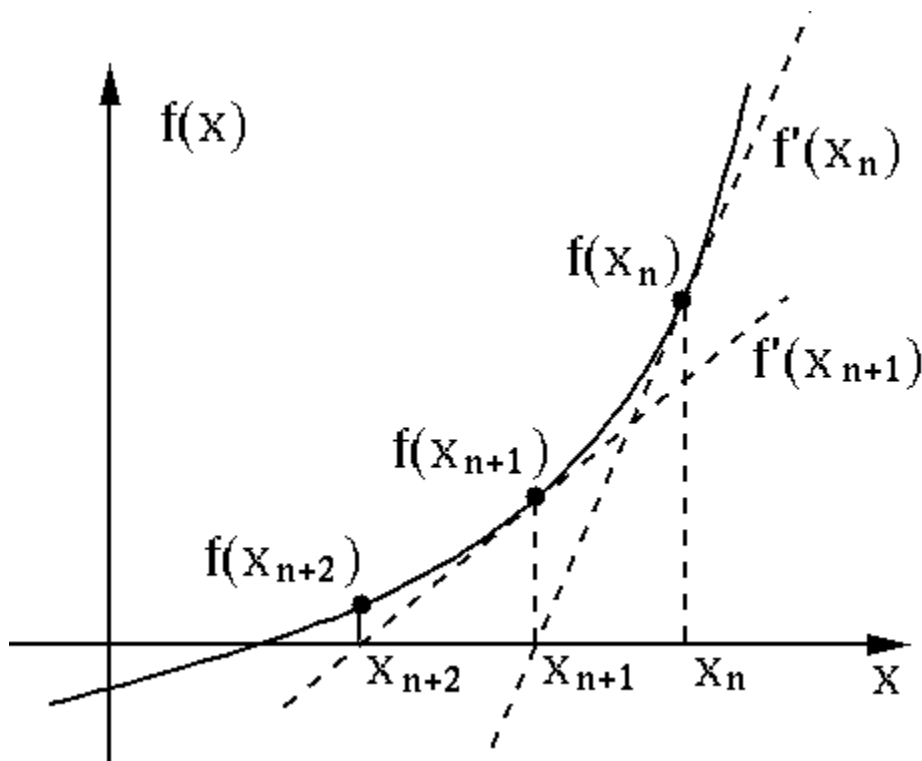
El método comienza con una función f definida sobre los números reales x , la derivada de la función f' y una estimación inicial x_0 para una raíz de la función f . Si la función satisface las suposiciones hechas en la derivación de la fórmula y la estimación inicial está cerca, entonces

una mejor aproximación x_1 es

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}$$

Geométricamente, $(x_1, 0)$ es la intersección del eje x y la tangente de la gráfica de f en $(x_0, f(x_0))$.

El proceso se repite como
$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$
, hasta que se alcanza un valor suficientemente preciso.



Aunque en general el método de Newton-Raphson es muy eficiente, hay situaciones donde se comporta de manera deficiente.

Por ejemplo, en el caso especial de raíces múltiples. Sin embargo, también cuando se trata de raíces simples, se encuentran dificultades, como por ejemplo la función.

$$f(x) = x^{10} - 1$$

v ?

Si se aplica el método se observará que, aunque la técnica converge a la raíz (1), lo hace muy lentamente, $n=\infty$, $X=1.00000$

Casos donde se presentan otras dificultades:

- Punto de inflexión [$F''(x)=0$], en la vecindad de una raíz.
- Tendencia del método a oscilar alrededor de un mínimo o un máximo local.
- valor inicial cercano a una raíz salta a una posición varias raíces más lejos. Esta tendencia a alejarse del área de interés se debe a que se encuentran pendientes cercanas a cero.
- Una pendiente cero [$f'(x) = 0$], causa una división entre cero en la fórmula de Newton-Raphson, lo cual ocasiona que la solución se dispare horizontalmente y jamás toca al eje x.

Uno comienza con una conjetura inicial que está razonablemente cerca de la raíz verdadera, luego la función se aproxima por su línea tangente (que se puede calcular utilizando las herramientas de cálculo), y uno calcula la intersección de x de esta línea tangente (que se hace fácilmente con álgebra elemental). Esta intersección con X generalmente será una mejor aproximación a la raíz de la función que la suposición original, y el método puede ser iterado.

El método de Newton es una técnica extremadamente poderosa; en general, la convergencia es cuadrática: a medida que el método converge en la raíz, la diferencia entre la

raíz y la aproximación es cuadrada (el número de dígitos exactos se duplica aproximadamente) en cada paso. Sin embargo, existen algunas dificultades con el método: dificultad para calcular la derivada de una función, falla del método para converger a la raíz, si no se cumplen los supuestos en la prueba de la convergencia cuadrática del método de Newton, la convergencia lenta para las raíces de multiplicidad mayor que 1.

Marco Teórico Del Método De Newton Raphson

Es un lenguaje de programación interpretado, multiparadigma y multiplataforma usado, principalmente, en Big Data, AI (Inteligencia Artificial), Data Science, frameworks de pruebas y desarrollo web. Esto lo convierte en un lenguaje de propósito general de gran nivel debido a su extensa biblioteca, cuya colección ofrece una amplia gama de instalaciones.

Python se gestó durante las vacaciones de Navidad de 1989, cuando el desarrollador holandés Guido van Rossum decidió escribir un intérprete para el nuevo lenguaje de scripting que venía trabajando.

Su amplia experiencia en la implementación del sistema ABC —un lenguaje de programación interactivo, estructurado y de alto nivel— se sumó a su iniciativa por crear un lenguaje más sencillo, intuitivo y potente. Así, en 1991, nació Python conocido en la actualidad como el sucesor del lenguaje ABC.

¿Cuáles son las características de Python?

El lenguaje Python se caracteriza por ser simple, rápido y tener una curva de aprendizaje amigable y corta. Está desarrollado bajo una licencia de código abierto, por lo que es de libre uso y distribución.

Pero ¿qué quiere decir “interpretado”, “multiparadigma” y “multiplataforma”? Te lo explicamos en sencillo:

- Interpretado: significa que Python “interpreta” el código del programador, es decir, lo traduce y lo ejecuta a la vez.
- Multiparadigma: porque es un lenguaje de programación que admite el uso de varios paradigmas de programación (modelos de desarrollo), por lo que no exige a los programadores un estilo único de programación. ¿Cuáles son los paradigmas de programación que permite Python? Programación orientada a objetos, programación imperativa y programación funcional.
- Multiplataforma: el lenguaje Python puede ejecutarse en diferentes sistemas operativos como Unix, Linux, macOS y Windows.

¿Qué se puede hacer con Python?

Una encuesta realizada en 2019 por el propio Python arrojó que el 59% de los desarrolladores usa Python para Data Analysis. Pero también para lo siguiente:

- Web Development
- Machine learning
- DevOps / System administration / Writing automation scripts

- Programing of web parsers / scrapers / crawlers
- Software testing / Writing automated tests

Metodología

En este apartado se encontrarán los pasos que se siguieron para la realización de este mini proyecto el cual consiste en realizar un software sobre un método matemático.

1. **Análisis:** adaptándonos a los requisitos o requerimientos de nuestro profesor y de acuerdo con las pautas que el mismo nos marca, analizamos cada uno de los puntos que nos indica para estudiar las diferentes formas de solución que existen para la problemática.

1.1. Primeramente, se realizó una breve investigación acerca del método de newton Raphson, cabe recalcar que fue breve, ya que previamente se había estudiado y comprendido dicho método, esta investigación se realizo para reforzar conceptos y demás factores para el desarrollo del software

1.2. Una vez realizado una breve investigación se procedió a analizar el lenguaje de programación Python, estudiando toda la teoría necesaria y todo el enfoque

matemático con el que necesitábamos correlacionar el método y el lenguaje, realizando consultas que nos ayudaran a entender el lenguaje Python.

2. **Compresión del lenguaje Python:** Una vez tenido la experiencia y el conocimiento necesario del método, se procedió a estudiar la sintaxis y la lógica del lenguaje Python, teniendo en cuenta que nunca habíamos trabajado con dicho lenguaje, fue una de las fases mas arduas en cuanto a tiempo y sacrificio, comprendiendo los diferentes entornos, librerías y demás que ofrecía Python, a pesar que no era un lenguaje 100% dirigido hacia la matemática, dicha compresión se fue guiada hacia la misma, tratando de programar el método en Python
3. **Propuesta:** después de realizar las respectivas investigaciones con sus debidos análisis, se desarrollaron varias propuestas con diferentes formas de solucionar el método con Python, de acuerdo a diferentes puntos de vista y argumentos de personas que ya conocían dicho lenguaje de programación, debido a que nosotros somos los desarrolladores principales somos las personas encargadas de mirar que solución es mas factible, teniendo en cuenta los requerimientos dados anteriormente por nuestro tutor, y teniendo en cuenta que el software debe ser amigable y comprendido por los usuarios que quisieran interactuar con dicho software matemático.
4. **Ejecución:** una vez estudiado al 100% el método de newton Raphson y comprendiendo la lógica y sintaxis que manejaba Python, se procedió a la ejecución del software teniendo en cuenta las propuestas realizadas en el ítem anterior, analizando cual era la mas factible. Y desarrollando el software

5. **Pruebas:** una vez desarrollado el software se procedió a realizar las diferentes pruebas necesarias para analizar el buen funcionamiento del mismo, analizando si los datos eran correctos y anotando los diferentes errores y bugs que se presentaron.
6. **Correcciones:** una vez realizadas las pruebas se procedieron a corregir todos los problemas que presentaba el software, tanto internos como visuales.
7. **Entrega del producto:** con todos los pasos realizados se procede a entregar el software cumpliendo con los requerimientos pedidos por nuestro profesor, y teniendo en cuenta algunas exigencias de los usuarios finales para el funcionamiento.

Resultados

Los resultados del software acerca de la interfaz y la lógica la podemos observar en la guía de usuario donde se detallan específicamente cada elemento y widgets del software.

A continuación, se mostrará una foto del software completo, y un ejemplo de un ejercicio donde se evidencia el funcionamiento del software.

Calculadora De Metodo de Newton-Raphson

ArchivoEditarGuia UsuariosAyudaContacto

17:54:44

Metodo de Newton Raphson

?

Esta calculadora en línea implementa el método de Newton (también conocido como el método de Newton-Raphson) usando la calculadora de derivadas para obtener una forma analítica de la derivada de la función dado, porque este método lo requiere. La teoría para recordar los conceptos básicos del método se puede encontrar debajo de la calculadora

Función $F(x)$:

Derivada de $F'(x)$

Calculadora De Derivadas

Derivar Funcion

Valor Inicial

Error Relativo

Resolver

Limpiar tabla

| i | x_i | $F(x_i)$ | $F'(x_i)$ | $ E_n $ |
|---|-------|----------|-----------|---------|
|---|-------|----------|-----------|---------|

Evidencia del ejercicio los datos del ejemplo son:

EJEMPLO 1

Función: e^{-x-x}

Derivada: $-1-e^{-x} \cdot \log(e)$ o $-e^{-x}-1$

Valor inicial de 0

Error relativo $\leq 0.1\%$

Calculadora De Metodo de Newton-Raphson

ArchivoEditarGuía UsuariosAyudaContacto

17:57:43

Metodo de Newton Raphson

?

Esta calculadora en línea implementa el método de Newton (también conocido como el método de Newton-Raphson) usando la calculadora de derivadas para obtener una forma analítica de la derivada de la función dada, porque este método lo requiere. La teoría para recordar los conceptos básicos del método se puede encontrar debajo de la calculadora

Función F(x):

e^x-x-x

Derivada de F(X)

Calculadora De Derivadas

Derivar Funcion

-1-e^x-x*log(e)

la derivada es -1 - e^x(-x)*log(e)

Valor Inicial

0

Error Relativo

0.001

Resolver

Limpia tabla

| i | x _i | F(x _i) | DF(x _i) | E _n |
|---|----------------|--------------------|---------------------|----------------|
|---|----------------|--------------------|---------------------|----------------|

Calculadora De Metodo de Newton-Raphson

ArchivoEditarGuía UsuariosAyudaContacto

17:57:58

Metodo de Newton Raphson

?

Esta calculadora en línea implementa el método de Newton (también conocido como el método de Newton-Raphson) usando la calculadora de derivadas para obtener una forma analítica de la derivada de la función dada, porque este método lo requiere. La teoría para recordar los conceptos básicos del método se puede encontrar debajo de la calculadora

Función F(x):

e^x-x-x

Derivada de F(X)

Calculadora De Derivadas

Derivar Funcion

-1-e^x-x*log(e)

la derivada es -1 - e^x(-x)*log(e)

Valor Inicial

Validation De Datos

?

Los Datos Ingresados Son Correctos?

1. La Funcion ingresada es e^x-x-x

2. La Derivada ingresada es -1-e^x-x*log(e)

3. El valor inicial es 0

4. El error relativo es 0.001

Si

No

Limpia tabla

| i | x _i | F(x _i) | DF(x _i) | E _n |
|---|----------------|--------------------|---------------------|----------------|
|---|----------------|--------------------|---------------------|----------------|

Calculadora De Metodo de Newton-Raphson
 Archivo Editar Guia Usuarios Ayuda Contacto

17:58:13

Metodo de Newton Raphson

Esta calculadora en línea implementa el método de Newton (también conocido como el método de Newton-Raphson) usando la calculadora de derivadas para obtener una forma analítica de la derivada de la función dada, porque este método lo requiere. La teoría para recordar los conceptos básicos del método se puede encontrar debajo de la calculadora

Función F(x):

Derivada de F(x) Calculadora De Derivadas Derivar Funcion

la derivada es $-1 - e^{(-x)} \cdot \log(e)$

Valor Inicial

Error Relativo

Resolver

Limpiar tabla

| i | x_i | F(x) | DF(x) | E _A |
|---|--------|--------|---------|----------------|
| 0 | 0.0000 | 1.0000 | -2.0000 | --- |
| 1 | 0.3000 | 0.1065 | -1.6065 | 100.0000% |
| 2 | 0.5663 | 0.0013 | -1.5676 | 11.7093% |
| 3 | 0.5671 | 0.0000 | -1.5671 | 0.1487% |
| 4 | 0.5671 | 0.0000 | -1.5671 | 0.0000% |

El valor aproximado del 0 de la función es 0.5671 Para un error relativo porcentual aproximado de 0.0000 Que es menor a 0.001

EJEMPLO 2

Evidencia del ejercicio los datos del ejemplo son:

Función: $x^3 - x - 1$

Derivada: $3x^2 - 1$

Valor inicial de 1

Error relativo $\leq 0.001\%$

Calculadora De Metodo de Newton-Raphson

ArchivoEditarGuia UsuariosAyudaContacto

18:00:59

Metodo de Newton Raphson

?

Esta calculadora en línea implementa el método de Newton (también conocido como el método de Newton-Raphson) usando la calculadora de derivadas para obtener una forma analítica de la derivada de la función dada, porque este método lo requiere. La teoría para recordar los conceptos básicos del método se puede encontrar debajo de la calculadora

Función F(x):

x**3-x-1

Derivada de F(X)

3*x**2-1

3*x**2-1

la derivada es 3*x**2 - 1

Calculadora De Derivadas

Derivar Funcion

Valor Inicial

1

Error Relativo

0.001

Resolver

Limpiar tabla

| I | X0 | F(X) | DF(X) | Ea |
|---|----|------|-------|----|
|---|----|------|-------|----|

Calculadora De Metodo de Newton-Raphson

ArchivoEditarGuia UsuariosAyudaContacto

18:01:15

Metodo de Newton Raphson

?

Esta calculadora en línea implementa el método de Newton (también conocido como el método de Newton-Raphson) usando la calculadora de derivadas para obtener una forma analítica de la derivada de la función dada, porque este método lo requiere. La teoría para recordar los conceptos básicos del método se puede encontrar debajo de la calculadora

Función F(x):

x**3-x-1

Derivada de F(X)

3*x**2-1

3*x**2-1

la derivada es 3*x**2 - 1

Calculadora De Derivadas

Derivar Funcion

Valor Inicial

Validation De Datos

?

Los Datos Ingresados Son Correctos?

1. La Funcion ingresada es x**3-x-1

2. La Derivada ingresada es 3*x**2-1

3. El valor inicial es 1

4. El error relativo es 0.001

SíNo

Limpiar tabla

| I | X0 | F(X) | DF(X) | Ea |
|---|----|------|-------|----|
|---|----|------|-------|----|

18:01:33

Metodo de Newton Raphson



Esta calculadora en línea implementa el método de Newton (también conocido como el método de Newton-Raphson) usando la calculadora de derivadas para obtener una forma analítica de la derivada de la función dada, porque este método lo requiere. La teoría para recordar los conceptos básicos del método se puede encontrar debajo de la calculadora

Función F(x):

Derivada de F(X)

Calculadora De Derivadas

Derivar Funcion

la derivada es $3*x**2 - 1$

Valor Inicial

Error Relativo

Resolver

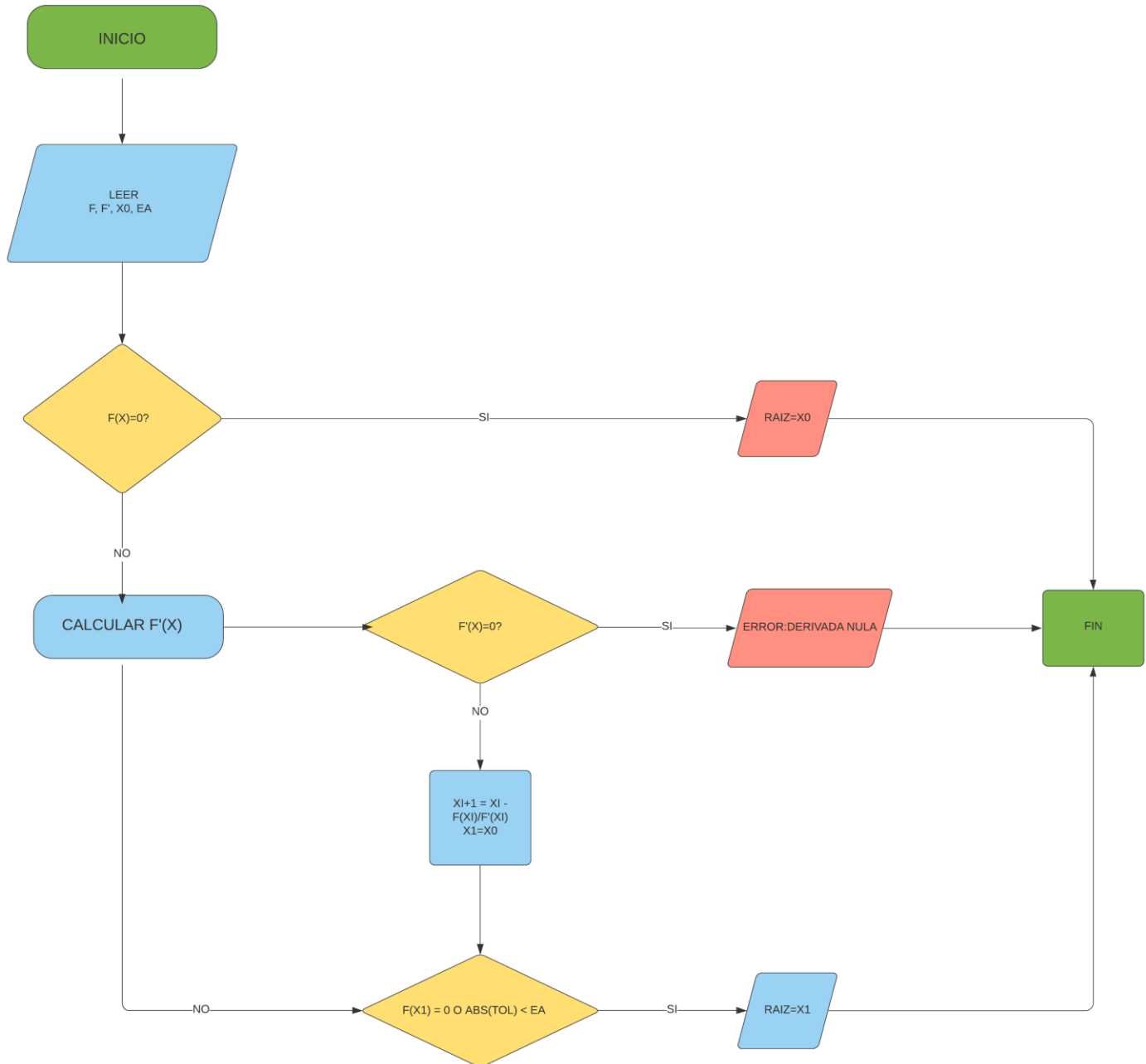
Limpiar tabla

| i | x_i | $F(x_i)$ | $F'(x_i)$ | $ E_n $ |
|---|--------|----------|-----------|----------|
| 0 | 1.0000 | -1.0000 | 2.0000 | --- |
| 1 | 1.5000 | 0.8750 | 5.7500 | 33.3333% |
| 2 | 1.3478 | 0.1007 | 4.4699 | 11.2603% |
| 3 | 1.3252 | 0.0021 | 4.2685 | 1.7075% |
| 4 | 1.3247 | 0.0000 | 4.2646 | 0.0364% |
| 5 | 1.3247 | 0.0000 | 4.2646 | 0.0000% |

El valor aproximado del 0 de la funcion es 1.3247 Para un error relativo porcentual aproximado de 0.0000 Que es menor a 0.001

Diagrama de flujo del software sobre el método de newton Raphson

Diagrama de Flujo Metodo Newton Raphson
ARMANDO PINEDA 191697



Código Fuente

Nombre del proyecto: Algoritmo De Newton Raphson Estándar

Versión: V.1

Descripción: The Newton-Raphson method

Fecha de creación: 26/04/2021

Creado por: Armando Pineda Paredes

Diseñado por: Armando Pineda Paredes

Comenta rizado por: Armando Pineda Paredes

Profesión: Estudiante

Universidad: Universidad Francisco de paula Santander Ocaña (ufpso)

Código Universitario:191697

E-MAIL: ajpineda@ufpso.edu.co -
armandojosepineda26@gmail.com

Ciudad: Ocaña Norte De Santander

Derechos reservados: Alcander.SA

Proyecto: Link de GitHub

```
from tkinter import * ## importamos la librería tkinter para la interfaz gráfica del programa
```

```
from tkinter import ttk ## hacemos el llamado de la librería y le asignamos la variable ttk
```

```
import tkinter as tk
```

```
from math import * ## importamos la librería math para operaciones matemática simples
```

```
import numpy as np ## importamos la librería numpy para la creación de arreglos y trabajar  
con ellos en operaciones matemáticas
```

```
from sympy import Derivative, diff, simplify, Symbol
```

```
import sympy as sp
```

```
import time
```

```
from tkinter import messagebox
```

```
from tkinter.messagebox import askyesno
```

```
from tkinter import filedialog
```

```
##-----Parte Grafica-----
```

```
raiz=Tk()                                ## Creamos un nombre variable igualado a la tk  
haciendo referencia a tkinter
```

```
raiz.title("Calculadora De Metodo de Newton-Raphson")  ## Con la variable raiz. creamos un  
titulo para la ventana de la interfaz
```

```
raiz.geometry('1920x1060')                ## Creamos un tamaño de 1920x1080 para que sea  
por defecto
```

```
raiz.configure(background='red',cursor="spraycan")    ## Configuramos la raiz, que va hacer  
el contenedor principal
```

```
raiz.iconbitmap("calculadora.ico")
```

```
Frame=Frame(raiz)                          ## Creamos un contenedor donde van a ir todos  
los datos, dentro de raiz
```


Frame.pack(fill="both",expand="True") ## le damos la propiedad de enpaquetar
y le decimos que sea algo tipo responsive

Frame.config(height="500",width="500",relief="ridge",bd=10) ## configuramos el contenedor
FRAME y le damos color de fondo y tamaño

clock=Label(Frame,font=("times",50,"bold"))

#f5f5f5

##imagine de fondo

##img = PhotoImage(file="instagram.gif") ## Creamos una variable llamada img y
hacemos el llamado del archivo con el file

##fondo = Label(Frame, image=img).place(x=0,y=0) ##creamoon label que representa el
fondo y le asignamos el nombre de la variable de la imagen

##-----Variables-----

error_relativo = StringVar() ## Creamos la variable error_relativo y le asignamos de tipo
double para que reciba números decimales

valor_inicialv = StringVar() ## Creamos la variable Valor_inicialv y le asignamos de
tipo double para que reciba numeros decimales

derivadaf = StringVar() ## Creamos la variable derivadaf de tipo StrinVar que es tipo
texto, ya que vamos a recibir ese dato como texto

funcion = StringVar() ## Creamos la variable funcion de tipo StrinVar que es tipo
texto, ya que vamos a recibir ese dato como texto

resultadoFuncion = StringVar() ## Creamos la variable resultado donde se mostrara
todo de tipo StrinVar que es tipo texto

##-----Parte Grafica-----

##titulo del proyecto

tituloLabel=Label(Frame, text="Metodo de Newton Raphson", font = ("Comic Sans MS", 16, "bold")) ## Creamos un label que muestra un texto que es "metodo de newton raphson" y le colocamos el tipo de letra, tamaño y en negrilla

tituloLabel.pack(padx=5,pady=5,ipadx=5,ipady=5) ##

Empaquetamos el titulo y le damos 5 espacio hacia la derecha,izquierda,arriba y abajo del label

tituloLabel.config(## Colocamos la etiqueta

config que sirve para configurar parametros de diseño del label

foreground="#fa7014", ## Le colocamos un

color de letra de color naranja ## Le colocamos

un color de fondo gris

justify="center" ## Le decimos que se

posicione en el centro

)

##introduccion para el metodo de newton rapshon

descripcionLabel=Label(Frame, text="\n Esta calculadora en línea implementa el método de Newton (también conocido como el método de Newton-Raphson) usando la calculadora de derivadas para obtener una forma analítica \n de la derivada de la función dada, porque este método lo requiere. La teoría para recordar los conceptos básicos del método se puede encontrar debajo de la calculadora", font = ("Comic Sans MS", 13))

descripcionLabel.pack(padx=5,pady=5,ipadx=5,ipady=5) ## Arriba

se crea el label para describir el metodo de newton y se le dan propiedades

```

descripcionLabel.config(                                ## Se configura el label de
descripcion con .config

    foreground="black",                                ## Le colocamos un
color de letra negro

    justify="left"                                     ## Le decimos que se
posicione a la izquierda

)

```

#label para la creacion del subtitulo "funcion f(x)"

```
funcionLabel=Label(Frame, text="Función F(x):", font = ("Comic Sans MS", 12, "bold"))
```

Creamos el label donde va el texto de funcion f(x)

```
funcionLabel.pack(padx=5,pady=5,ipadx=5,ipady=5)                                ##
```

Empaquetamos el titulo y le damos 5 espacio hacia la derecha,izquierda,arriba y abajo del label

```
funcionLabel.config(                                ## Se configura el label de
descripcion con .config
```

```
    foreground="#fa7014",                                ## Le colocamos un
color de letra naranja
```

```
    justify="left"                                     ## Le decimos que se
ajuste a la izquierda en su cuadro de texto
```

```
)
```

##caja de texto que va al lado del subtitulo "funcion f(x)""

```
cuadroFuncion=Entry(Frame,textvariable=funcion, font = ("Comic Sans MS", 10, "bold"))  
## Creamos un entry que sera por donde entran los datos, esepificamente donde entran los datos  
del label anterior que es funcion
```

```
cuadroFuncion.pack(padx=5,pady=5,ipadx=5,ipady=5) ##  
Empaquetamos el titulo y le damos 5 espacio hacia la derecha,izquierda,arriba y abajo del label
```

```
##subtitulo para la creacion de "derivada de f(x)"
```

```
derLabel=Label(Frame, text="Derivada de F(X)", font = ("Comic Sans MS", 12, "bold") )  
## Creamos el texto que diga Derivada de f(x) y le colocamos tipo de fuente,tama{o y negrilla}
```

```
derLabel.pack(padx=5,pady=5,ipadx=5,ipady=5) ##  
Empaquetamos el titulo y le damos 5 espacio hacia la derecha,izquierda,arriba y abajo del label
```

```
derLabel.config( ## Se configura el label de  
descripcion con .config
```

```
    foreground="#fa7014", ## Le colocamos un  
color de letra naranja
```

```
    justify="left" ## Le decimos que se  
ajuste a la izquierda en su cuadro de texto
```

```
)
```

```
##caja de texto que reprenta al subtitulo "derivada de f(x)"
```

```
cuadroder=Entry(Frame,textvariable=derivadaf, font = ("Comic Sans MS", 10, "bold"))  
## Creamos la caja de texto para que el usuario ingrese la derivada
```

```
cuadroder.pack(padx=5,pady=5,ipadx=5,ipady=5)          ## Lo
empaquetamos y le damos propiedades
```

```
##label que representa el subtitulo "valor inicial"
```

```
vlLabel=Label(Frame, text="Valor Inicial", font = ("Comic Sans MS", 12, "bold"))
```

```
## Creamos un label donde mostrar el nombre de valor inicial
```

```
vlLabel.pack(padx=5,pady=5,ipadx=5,ipady=5)          ## Creamos
una configuracion para esa caja de texto
```

```
vlLabel.config(
```

```
    foreground="#fa7014",                               ## Le damos un
color de letra de #fa7014
```

```
    justify="left"                                       ## y que cuando empiece
a escribir, las letras aparezcan desde la izquierda
)
```

```
##caja de texto que pertenece al subtitulo "valor inicial"
```

```
cuadrovl=Entry(Frame,textvariable=valor_inicialv, font = ("Comic Sans MS", 10, "bold"))
```

```
## Creamos la respectiva caja de texto donde el usuario ingresara ese valor inicial
```

```
cuadrovl.pack(padx=5,pady=5,ipadx=5,ipady=5)          ## Le damos
propiedades de posicionamiento a esa caja donde el usuario va a escribir
```

```
##creacion del label para el subtitulo "error relativo"
```

```
errorLabel=Label(Frame, text="Error Relativo", font = ("Comic Sans MS", 12, "bold"))
```

```
## Creamos un label donde va a mostrar el mensaje error relativo
```

```
errorLabel.pack(padx=5,pady=5,ipadx=5,ipady=5)                ## le damos  
una configuracion para que se ubique correctamnete a ese label de mensaje
```

```
errorLabel.config(  
    ## Le empezamos a dar una caonfiguracion  
    foreground="#fa7014",                ## Le colocamos  
    color de letras naranja a ese mensaje que aparece  
    justify="left"  
    ## y que cuando empiece a escribir, las letras aparescan  
    desde la izquierda  
)
```

```
##creacion de la caja de texto para el subtitulo "error relativo"
```

```
cuadroerror=Entry(Frame,textvariable=error_relativo, font = ("Arial", 10, "bold"))    ##  
Creamos la caja de texto que es donde va a escribir el eror relativo, le asignamos la valiebra y le  
damos propiedades de letra, tamaño y negrilla  
cuadroerror.pack(padx=5,pady=5,ipadx=5,ipady=5)
```

```
errorLabel=Label(Frame, text="Calculadora De Derivadas", font = ("Comic Sans MS", 12,  
"bold"))    ## Creamos un label donde va a mostrar el mensaje error relativo
```

```
errorLabel.place(x=1050,y=257)                ## le damos una configuracion  
para que se ubique correctamnete a ese label de mensaje
```

```
errorLabel.config(  
    ## Le empezamos a dar una caonfiguracion  
    foreground="#fa7014",                ## Le colocamos  
    color de letras naranja a ese mensaje que aparece
```

```
justify="left"

## y que cuando empiece a escribir, las letras aparezcan
desde la izquierda
```

```
) ## y le damos uss respectivas propiedades
```

```
##Mostrar lo ingresado
```

```
Mostrar_Funcion= StringVar()
```

```
Mostrar_Derivada= StringVar()
```

```
Mostrar_ValorI= StringVar()
```

```
Mostrar_Error= StringVar()
```

```
Resultado_Mio= StringVar()
```

```
TextoRespuesta=Label(raiz,text="el resultado es: ",textvariable=Resultado_Mio,font=("",12))
```

```
TextoRespuesta.place(x=500,y=870)
```

```
##-----Operaciones Matematicas Del Metodo Newton
```

```
Raphson-----
```

```
rest = StringVar()
```

```
##creacion del label para mostrar derivada
```

```
textoR = Label(Frame,text="el resultado es: ",textvariable=rest, font = ("Comic Sans MS", 12))
```

```
textoR.place(x=1050,y=303)
```

##funcion que recibe y realiza todas las operaciones

def mostrar():

 answer=messagebox.askyesno('Validacion De Datos', 'Los Datos Ingresados Son
Correctos? '+'\n1. La Funcion ingresada es '+ str(funcion.get()) +'\n2. La Derivada ingresada es '
+ str(derivadaf.get())+'\n3. El valor inicial es ' + str(valor_inicialv.get()) + '\n4. El eror relativo es
' + str(error_relativo.get()))

 if answer:

 ## Creamos la

funcion con el nombre de "mostrar()", que hace que solo se active esa funcion solo si se activa el
boton que se llama de la misma forma "mostrar"

 f = lambda x:eval(funcion.get())

 ## Llamamos la caja de texto

(entry) llamada funcion y la convertimos de texto a matematica y la guardamos en la variable f

 df = lambda x:eval(derivadaf.get())

 ## Hacemos el llamado de la caja de

texto (entry) derivada f y la convertimos en matematica y la guardamos en la variable df

 xi = eval(valor_inicialv.get())

 ## Hacemos el llamado de la

caja de texto (entry) valor inicial y la convertimos en matematica y la guardamos en la variable
xi

 tol = eval(error_relativo.get())

 ## Hacemos el llamado de la caja de

texto (entry) error relativo y la convertimos en matematica y la guardamos en la variable tol
(haciendo referencia a tolerancia o erroe relativo)


```
tramo = abs(tol)
```

```
## La variable tol le
```

colocamos abs(tol) para que la tolerancia ingresada se convierta en valor absoluto, siempre positiva, nunca un valor negativo y la guardamos en la variable tramo

```
x = xi
```

```
## la variable xi
```

luego de transformada la guardamos en la variable x

```
print("valores del problema")
```

```
print('* la funcion ingresada es =',(funcion.get()))
```

```
print('* la derivada ingresada es =',derivadaf.get())
```

```
print('* el valor inicial es x0 =',xi)
```

```
print('* la error es =',tol)
```

```
print('-----')
```

```
iteraciones=100
```

```
c=0
```

```
xprimero=x
```

```
tv.insert(",20,values=(c,'{: .4f}'.format(xprimero),'{: .4f}'.format(f(xprimero)),'{: .4f}'.format(df(xprimero)),'---'))
```

```
while ( tramo >= tol and c <= iteraciones):
```

```
## Creamos un while para realizar todas las
```

iteraciones solo si el tramo(error relativo) es mayor o igual al tol(valor inicial) que es lo que el usuario ingresa

$$x_{\text{nuevo}} = (x) - (f(x)/df(x))$$

Aplicamos la formula para sacar x_{i+1} y la guardamos en una variable llamada x_{nuevo} que hace referencia a x_i siguiente

$$\text{tramo} = \text{abs}(((x_{\text{nuevo}} - x)/x_{\text{nuevo}}) * 100)$$

Aplicamos la formula para calcula el error relativo y con el abs() lo convertimos en

```
##tabla.append([x,xnuevo,tramo])
```

```
x = xnuevo
```

```
c+=1
```

Aca guaradamos a x_{nuevo} de nuevo en la variable de x y es como si e usuario estuviera ingresando de nuevo el x_i , y asi realiza todas las iteraciones

```
tv.insert('20,values=(c, '{:.4f}'.format(x), '{:.4f}'.format(f(x)), '{:.4f}'.format(df(x)), '{:.4f}'.format(tramo) + '%'),tags=('BUY'))
```

Aca solo nos encargamos de llamar todas las variables para imprimirlas, y como esta dentro del while cada ves que ahce las iteraciones las arroja, y le decimos que las arroje con 4 decimales depues del punto

```
print('\nxi: {:.4f}'.format(x))
```

Este comando es para imprimir la x inicial pero en consola, esta zona el usuario no la ve, es solo para llevar un control del funcionamiento

```
print('con un error relativo de: {:.4f}'.format(tramo) + '%')
```

Este comando es para imprimir la error relativo porcentual pero en consola, esta zona el usuario no la ve, es solo para llevar un control del funcionamiento

```
print('f(x): {:.4f}'.format(f(xnuevo)))
```

Este comando es para imprimir la $F(X_i)$ pero en consola, esta zona el usuario no la ve, es solo para llevar un control del funcionamiento

```
print('f(xi): {:.4f}'.format(df(xnuevo)))
```

Este comando es para imprimir la DF(XI) inicial pero en consola, esta zona el usuario no la ve, es solo para llevar un control del funcionamiento

##('el valor aproximado del 0 de la funcion es: '{:.4f}'.format(x),'para un error relativo porcentual aproximado de : {:.4f}'.format(tramo),' que es menor a: ',tol)

res=str(('El valor aproximado del 0 de la funcion es {:.4f}'.format(x) + ' Para un error relativo porcentual aproximado de {:.4f}'.format(tramo) + ' Que es menor a ' + str(tol))) ## Aca solo estamos imprimiendo las respuesta oficial y haciendo el tamaño de los componenten como: xi, el porcentaje y hacemos alucion que ese porcentaje es menor al error ingresado por el usuario, y lo guardamos en una variable res(resultado)

Resultado_Mio.set(res)

funcion.set("")

derivadaf.set("")

valor_inicialv.set("")

Hacemos el llamado de la caja de texto y dentro del parentesis le damos el valor de la variable res (resultado)

error_relativo.set("")

def Limpiar ():

rest.set("")

Resultado_Mio.set("")

for borrar in tv.get_children():

tv.delete(borrar)

```
def Derivar():
```

```
    x=Symbol('x')
```

```
    n=funcion.get()
```

```
    derivada=diff(n,x)
```

```
    rest.set("la derivada es " + str(derivada))
```

```
##-----Parte Grafica Botom-----  
-----
```

```
##boton
```

```
botonMostrar=Button(Frame, text="Resolver",font = ("MS Sans Serif", 11, "bold"),
```

```
command=mostrar, bg="#fa7014", fg = "#E6E9ED")          ## Creamos el boton llamado
```

```
resolver y lo guardamos en un comando que se llame mostrar, que es presionar este boton ejecuta  
todas las lienas de codigo anteriores
```

```
botonMostrar.pack(padx=5,pady=5,ipadx=5,ipady=5)
```

```
botonLimpiar=Button(Frame, text="Limpiar tabla",font = ("MS Sans Serif", 11, "bold"),
```

```
command=Limpiar, bg="#fa7014", fg = "#E6E9ED")          ## Creamos el boton llamado
```

```
resolver y lo guardamos en un comando que se llame mostrar, que es presionar este boton ejecuta  
todas las lienas de codigo anteriores
```

```
botonLimpiar.pack(padx=5,pady=5,ipadx=5,ipady=5)
```

```
botonDerivar=Button(Frame, text="Derivar Funcion",font = ("MS Sans Serif", 11, "bold"),
command=Derivar, bg="#fa7014", fg = "#E6E9ED")          ## Creamos el boton llamado
resolver y lo guardamos en un comando que se llame mostrar, que es presionar este boton ejecuta
todas las lienas de codigo anteriores
```

```
botonDerivar.place(x=1280,y=257)
```

```
botonDerivar.config(width="14",height="1")
```

```
## Le damos unas propiedaes de
posicionamiento
```

```
##-----Parte Grafica Tabla-----
-----
```

```
##tabla
```

```
tv=ttk.Treeview(raiz,columns=('T','XI','F(XI)','DF(XI)','|EA|'), show='headings')
```

```
tv.column('T',minwidth=0, width=85,anchor="n")          ## Creamos una tabla y le
definimos las columnas que va a tener y lo guardamos en TV
```

```
tv.column('XI',minwidth=0, width=85,anchor="n")          ##
Creamos la columna llamada XI y le damos propiedades de tamaño
```

```
tv.column('F(XI)',minwidth=0, width=105,anchor="n")      ##
Creamos la columna llamada F(XI) y le damos propiedades de tamaño
```

```
tv.column('DF(XI)',minwidth=0, width=105,anchor="n")      ##
Creamos la columna llamada DF(XI) y le damos propiedades de tamaño
```

```
tv.column('|EA|',minwidth=0, width=105,anchor="n")
```

```
tv.heading('T',text='T')          ## Creamos la columna llamada EA(error
porcentual) y le damos propiedades de tamaño
```

```
tv.heading('XI',text='XI')                                ## Ahora aca le  
estamos asignando una variable a esas columnas para poder identificar quien es quien y a la hora  
de mostrar saber donde mostrar, aca le asignamos XI que tenga un texto de XI
```

```
tv.heading('F(XI)',text='F(XI)')                          ## Ahora aca le  
estamos asignando una variable a esas columnas para poder identificar quien es quien y a la hora  
de mostrar saber donde mostrar, aca le asignamos F(XI) que tenga un texto de F(XI)
```

```
tv.heading('DF(XI)',text='DF(XI)')                       ## Ahora aca le  
estamos asignando una variable a esas columnas para poder identificar quien es quien y a la hora  
de mostrar saber donde mostrar, aca le asignamos DF(XI) que tenga un texto de DF(XI)
```

```
tv.heading('|EA|',text='|EA|')                            ## Ahora aca le  
estamos asignando una variable a esas columnas para poder identificar quien es quien y a la hora  
de mostrar saber donde mostrar, aca le asignamos EA que tenga un texto de EA(ERROR  
RELATIVO)
```

```
tv.place(x=710,y=640)                                    ## Aca le estamos  
dando donde se debe posicionar esa tabla en la interfaz
```

```
##boton crear nueva ventana
```

```
def v2():
```

```
    raiz2 = Toplevel()
```

```
    raiz2.geometry("1100x900")
```

```
    raiz2.title("Guia De Usuario")
```

```
    raiz2.iconbitmap("libro.ico")
```

```
##imagen del boton
```

```
imgBoton=PhotoImage(file="rueda.gif")
```

```
##boton
```

```
botonabrirventana=Button(Frame, text="guia de usuario",image=imgBoton,font = ("MS Sans  
Serif", 11, "bold"),command=v2,borderwidth="5",relief="flat")
```

```
botonabrirventana.place(x=1800,y=40)
```

```
botonabrirventana.config(
```

```
    width="40",
```

```
    height="40"
```

```
)
```

```
##reloj
```

```
def times():
```

```
    current_time=time.strftime("%H:%M:%S")
```

```
    clock.config(text=current_time,fg="#fa7014",font=("Helvetica",17))
```

```
    clock.after(1000,times)
```

```
clock.place(x=0,y=5)
```

```
times()
```

```
##menu
```

```
menubar = Menu(raiz)
```

```
raiz.config(menu=menubar)
```

```
filemenu = Menu(menubar, tearoff=0)
```

```
def NuevoA ():
```

```
    for borrar in tv.get_children():
```

```

        tv.delete(borrar)

    funcion.set("")

    derivadaf.set("")

    valor_inicialv.set("")
## Hacemos el llamado de la caja de texto y dentro del parentesis le damos el valor de la variable
res (resultado)

    error_relativo.set("")

    Mostrar_Funcion.set("")

    Mostrar_Derivada.set("")

    Mostrar_ValorI.set("")

    Mostrar_Error.set("")

    Resultado_Mio.set("")

filemenu.add_command(label="Nuevo Archivo",command=NuevoA)

filemenu.add_separator()

def AbrirA():

    print(filedialog.askopenfilename(initialdir = "/",title = "Abrir
Archivo",defaultextension=".txt",filetypes = (("Python files","*.py;*.pyw"),("All files","*.*"))))

filemenu.add_command(label="Abrir Archivo",command=AbrirA)

filemenu.add_separator()

def GuardarA():

    print(filedialog.asksaveasfilename(initialdir = "/",title = "Guardar Archivo",filetypes =
(("Python files","*.py;*.pyw"),("All files","*.*"))))

filemenu.add_command(label="Guardar Archivo",command=GuardarA)

```



```
filemenu.add_separator()
```

```
def clicked():
```

```
    answer=messagebox.askyesno('Advertencia', 'Seguro que quieres salir?')
```

```
    if answer:
```

```
        raiz.destroy()
```

```
filemenu.add_command(label="Salir",command=clicked)
```

```
editmenu = Menu(menubar, tearoff=0)
```

```
editmenu.add_command(label="Cortar")
```

```
editmenu.add_separator()
```

```
editmenu.add_command(label="Copiar")
```

```
editmenu.add_separator()
```

```
editmenu.add_command(label="Pegar")
```

```
helpmenu = Menu(menubar, tearoff=0)
```

```
def VAyuda():
```

```
    raiz2 = Toplevel()
```

```
    raiz2.geometry("500x600")
```

```
    raiz2.title("Ayuda De Usuario")
```

```
    raiz2.iconbitmap("libro.ico")
```

```
helpmenu.add_command(label="Ayuda",command=VAyuda)
```

```
helpmenu.add_separator()
```

```
def AcercaD():
```

```
    raiz3 = Toplevel()
```

```
    raiz3.geometry("900x639")
```

```
    raiz3.title("Acerca Del software")
```

```
    raiz3.iconbitmap("libro.ico")
```

```
    fondoacercad=PhotoImage(file="picasoacercade.gif")
```

```
    fondo7 = Label(raiz3, image=fondoacercad).place(x=0,y=0)
```

```
    raiz3.mainloop()
```

```
helpmenu.add_command(label="Acerca de...",command=AcercaD)
```

```
guiamenu = Menu(menubar, tearoff=0)
```

```
def GuiaM():
```

```
    raiz4 = Toplevel()
```

```
    raiz4.geometry("1200x675")
```

```
    raiz4.title("Guia De Metodo Newton-Raphson")
```

```
    raiz4.iconbitmap("libro.ico")
```

```
    raiz4.config(bg="white")
```

```
    fondoguiame=PhotoImage(file="picasometodo2.gif")
```

```
    fondo7 = Label(raiz4, image=fondoguiame).place(x=0,y=0)
```

```
    raiz4.mainloop()
```

```
guiamenu.add_command(label="Guia del metodo",command=GuiaM)
```

```
guiamenu.add_separator()
```

```
guiamenu.add_command(label="Guia del software",command=v2)
```

```
guiamenu.add_separator()
```

```
def EntradaD():
```

```
    raiz6 = Toplevel()
```

```
    raiz6.geometry("932x550")
```

```
    raiz6.title("Guiar Para Entrada De Los Datos")
```

```
    raiz6.iconbitmap("entradad.ico")
```

```
    raiz6.config(bg="white")
```

```
    Titulodatos = Label(raiz6,text="Operadores Matematicos Para La Entrada Del Software",  
font = ("Comic Sans MS", 13,"bold"))
```

```
    Titulodatos.place(x=250,y=0)
```

```
    Titulodatos.config(foreground="#fa7014",bg="white")
```

```
    Definiciondatos = Label(raiz6,text="En cuanto a los operadores aritméticos, estos  
permiten realizar las diferentes operaciones aritméticas del álgebra: suma, resta,",font = ("Comic  
Sans MS", 12))
```

```
    Definiciondatos.place(x=0,y=56)
```

```
    Definiciondatos.config(bg="white")
```

```
    Definiciondatos = Label(raiz6,text="producto, división, ...Estos operadores de Python  
son de los más utilizados. El listado completo es el siguiente:", font = ("Comic Sans MS", 12))
```

```
    Definiciondatos.place(x=0,y=80)
```

```
    Definiciondatos.config(bg="white")
```

```
    fondoentradaD=PhotoImage(file="picasotabla.gif")
```

```
fondo6 = Label(raiz6, image=fondoentradaD).place(x=140,y=129)

raiz6.mainloop()

guiamenu.add_command(label="Guia para entrada de datos",command=EntradaD)

guiamenu.add_separator()
```

```
def AnalisisD():
```

```
    raiz7 = Toplevel()

    raiz7.geometry("1200x675")

    raiz7.title("Analisis De Los Resultados")

    raiz7.iconbitmap("libro.ico")

    fondoanalisisd=PhotoImage(file="picasodatosm.gif")

    fondo6 = Label(raiz7, image=fondoanalisisd).place(x=0,y=0)

    raiz7.mainloop()

    guiamenu.add_command(label="Analisis de los datos",command=AnalisisD)

    contacmenu = Menu(menuubar, tearoff=0)
```

```
def DatosDe():
```

```
    raiz8 = Toplevel()

    raiz8.geometry("720x500")

    raiz8.title("Datos De Desarrollador")

    raiz8.iconbitmap("libro.ico")

    fondodesa=PhotoImage(file="picasodesarrollador.gif")
```

```

fondo2 = Label(raiz8, image=fondodesa).place(x=0,y=0)

raiz8.mainloop()

contacmenu.add_command(label="Desarrollador",command=DatosDe)

contacmenu.add_separator()

def DatosDi():

    raiz9 = Toplevel()

    raiz9.geometry("720x500")

    raiz9.title("Datos De Diseñador")

    raiz9.iconbitmap("libro.ico")

    fondodis=PhotoImage(file="picasodiseñador.gif")

    fondo2 = Label(raiz9, image=fondodis).place(x=0,y=0)

    raiz9.mainloop()

contacmenu.add_command(label="Diseñador",command=DatosDi)

contacmenu.add_separator()

def DatosMa():

    raiz10 = Toplevel()

    raiz10.geometry("720x500")

    raiz10.title("Datos Del Matematico")

    raiz10.iconbitmap("libro.ico")


    fondoma=PhotoImage(file="picasomatematico.gif")

    fondo3 = Label(raiz10, image=fondoma).place(x=0,y=0)

    raiz10.mainloop()

```

```
contacmenu.add_command(label="Matematico",command=DatosMa)
```

```
contacmenu.add_separator()
```

```
def DatosRe():
```

```
    raiz11 = Toplevel()
```

```
    raiz11.geometry("720x500")
```

```
    raiz11.title("Redes")
```

```
    RedeLabel=Label(raiz11, text="adsadsadsads", font = ("Comic Sans MS", 16, "bold"))
```

```
    RedeLabel.place(x=100,y=200)
```

```
    raiz11.iconbitmap("libro.ico")
```

```
    fondoBoton=PhotoImage(file="instagram.gif")
```

```
    fondo1=PhotoImage(file="picasoredes.gif")
```

```
    fondo = Label(raiz11, image=fondo1).place(x=0,y=0)
```

```
    ##botonabrirventana=Button(raiz11, text="mouse",image=fondoBoton,font = ("MS Sans  
Serif", 11, "bold"),borderwidth="5")
```

```
    ##botonabrirventana.place(x=100,y=40)
```

```
    raiz11.mainloop()
```

```
contacmenu.add_command(label="Redes",command=DatosRe)
```

```
contacmenu.add_separator()
```

```
def DatosCor():
```

```
    raiz12 = Toplevel()
```

```
raiz12.geometry("720x500")

raiz12.title("Datos De Correo")

raiz12.iconbitmap("libro.ico")

fondodesa=PhotoImage(file="picasocorreo.gif")

fondo4 = Label(raiz12, image=fondodesa).place(x=0,y=0)

raiz12.mainloop()

contacmenu.add_command(label="Correo",command=DatosCor)

menubar.add_cascade(label="Archivo", menu=filemenu)

menubar.add_cascade(label="Editar", menu=editmenu)

menubar.add_cascade(label="Guia Usuarios", menu=guiamenu)

menubar.add_cascade(label="Ayuda", menu=helpmenu)

menubar.add_cascade(label="Contacto", menu=contacmenu)

raiz.mainloop()  ## Esta es una de las partes mas importantes de la interfaz, cierra todo y une
todo las lineas, todo lo que se encuentre despues de este comando no lo interpreta como interfaz
grafica de tkinter
```

Guía de usuario

Desarrollo de la guía de usuario del software que calcula el método de newton Raphson

1. Menú del software

El software contara con un menú el cual es desplegable, ofreciéndole diferentes funcionalidades, para las ayudas del método

20:51:09

Metodo de Newton Raphson

Esta calculadora en línea implementa el método de Newton (también conocido como el método de Newton-Raphson) usando la calculadora de derivadas para obtener una forma analítica de la derivada de la función dada, porque este método lo requiere. La teoría para recordar los conceptos básicos del método se puede encontrar debajo de la calculadora

Función $F(x)$:

Derivada de $F(X)$ Calculadora De Derivadas Derivar Funcion

Valor Inicial

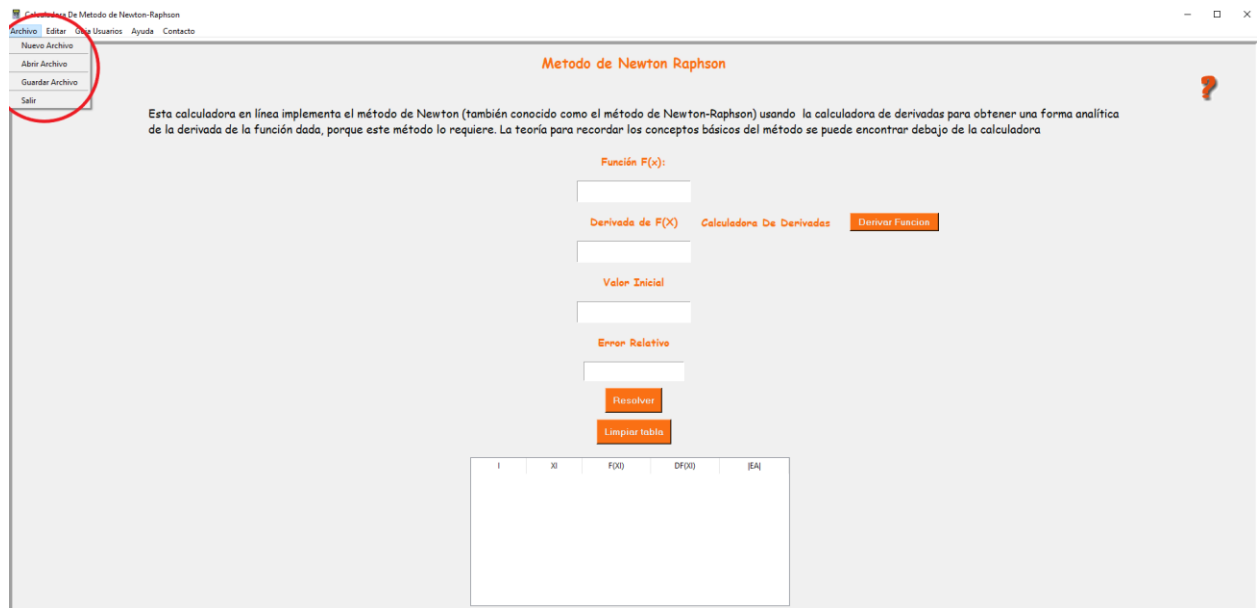
Error Relativo

Resolver

Limpiar tabla

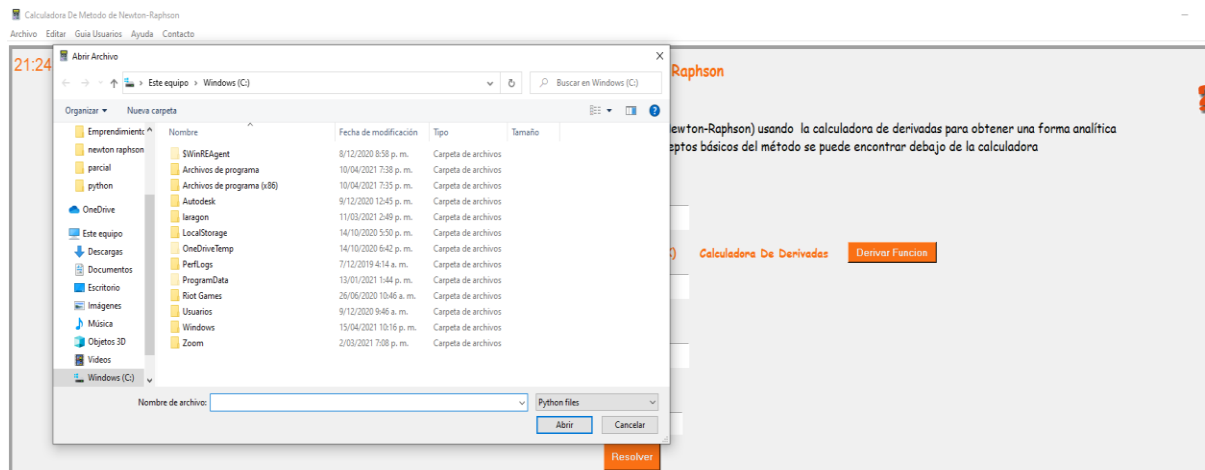
| I | X_i | $F(X_i)$ | $DF(X_i)$ | $ E_a $ |
|---|-------|----------|-----------|---------|
|---|-------|----------|-----------|---------|

1.1.Menú del software – Archivo: el software contara con un apartado de archivo, contando con 4 mini submenús que cada uno tiene sus respectivas funcionalidades.

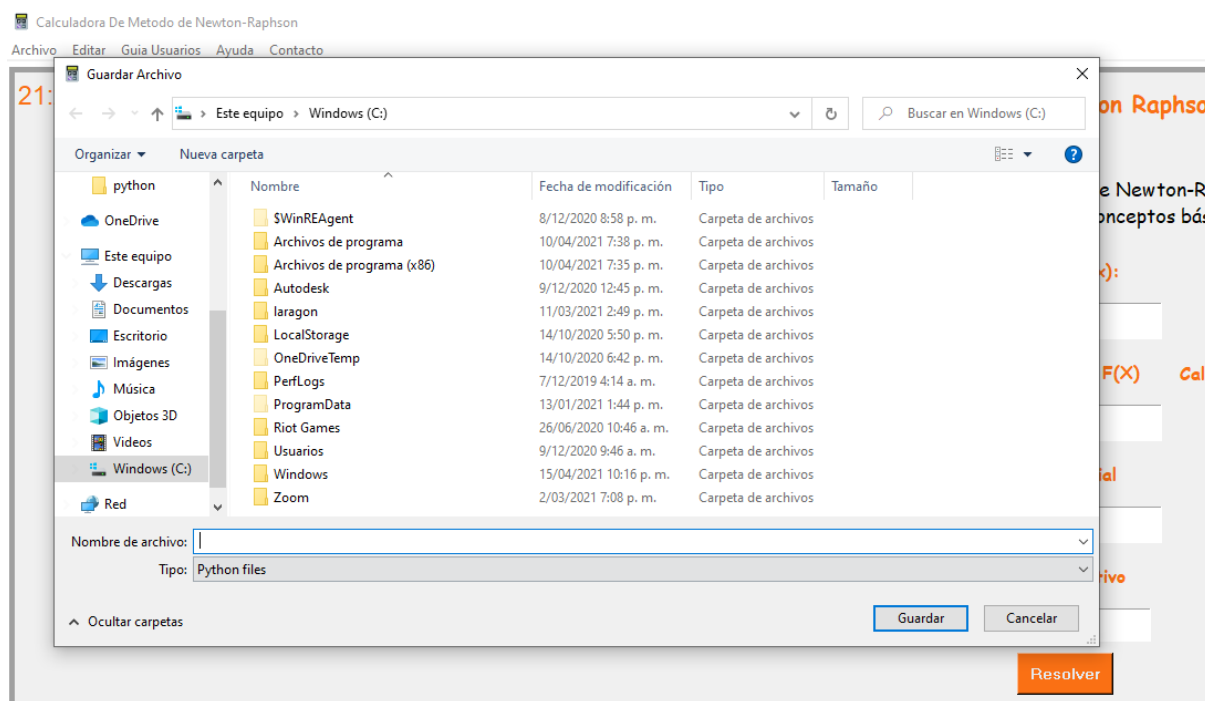


1.1.1 Archivo – Nuevo archivo: este submenú al presionarse el software borrara todos los campos en caso que estén llenos para darle al usuario la posibilidad de ingresar de nuevo los datos y así crear un nuevo archivo

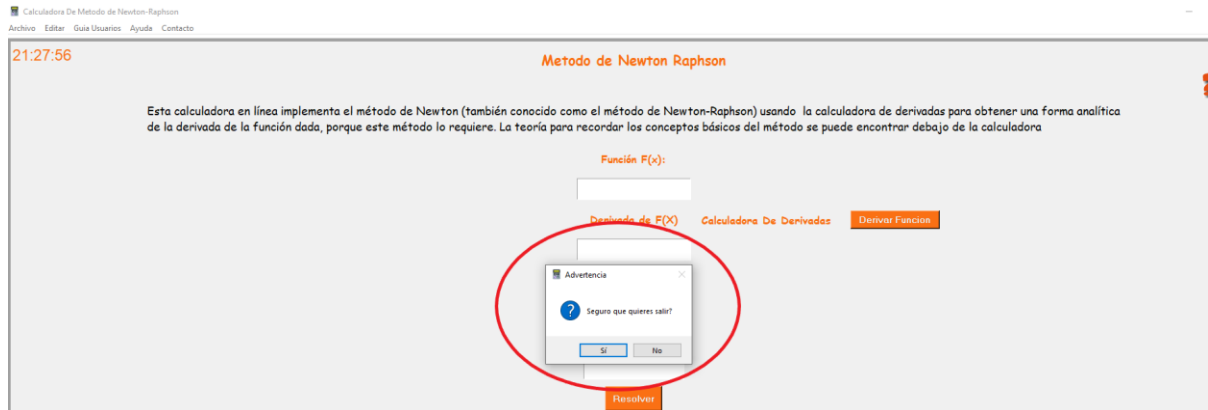
1.1.2 Archivo – Abrir archivo: en este apartado el usuario tendrá la posibilidad de abrir un archivo txt en donde podrá llenar los campos que se piden abriendo solamente el archivo, cabe recalcar que cuando el usuario cargue un archivo tendrá que darle al botón de resolver.



1.1.3 Archivo – Guardar archivo: el software le permitirá al usuario guardar el ejercicio resuelto en un archivo .txt, cabe recalcar que el software solo guardara el archivo si los campos están llenos y pudo realizar con satisfacción el ejercicio.

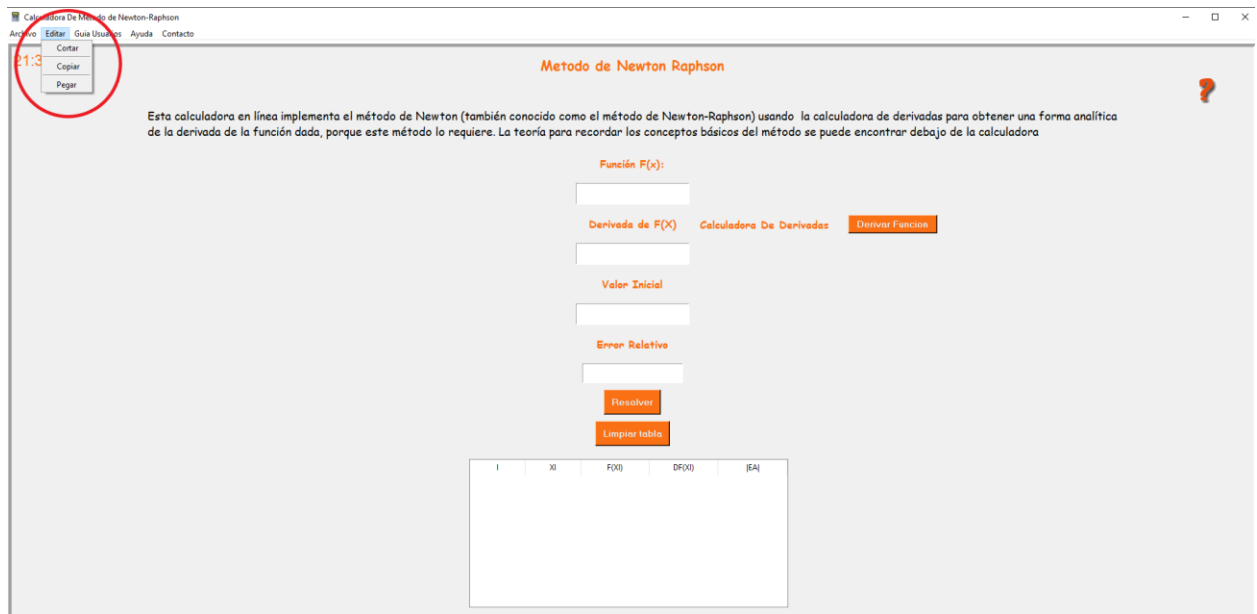


1.1.4 Archivo – Salir: el usuario al presionar este submenú tendrá la posibilidad de salir del software, preguntándole si desea salir o no.



1.2. Menú del software – Editar

El software le permitirá al usuario hacer acciones de editar, copiar y pegar ejercicios que ya halla realizado así ahorrando tiempo y no teniendo que copiar los ejercicios de nuevo



1.3. Menú del software – Guía usuarios

En este apartado el usuario podrá verificar o examinar las diferentes guías que existen para que el usuario se apoye en ellas.



1.3.1. Guía de usuarios – Guía del método: El usuario podrá consultar una guía del método de newton Raphson, para poder verificar conceptos y/o formulas para el buen entendimiento del método.

Guia De Metodo Newton-Raphson

Newton-Raphson

El Método Newton-Raphson es uno de los mas utilizados para localizar raíces ya que en general es muy eficiente y siempre converge para una función polinomial. Se obtiene a partir de un valor inicial x_i , desde el cual se puede trazar una tangente desde el punto $[x_i, f(x_i)]$ de la curva. Normalmente, el punto donde esta tangente cruza el eje x representa una aproximación a la raíz de la función.

En qué consiste el método

El método consiste en partir de una estimación inicial de la solución x_0 , a partir de la cual se va a ir construyendo una sucesión de aproximaciones mediante la formula.

La formula de Newton-Raphson es la siguiente:

$$x_{j+1} = x_j - \frac{f(x_j)}{f'(x_j)}.$$

El punto inicial mas uno es igual a el pinto inicial menos el punto inicial evaluado en la función entre el punto inicial evaluado en la derivada de la función.

El método comienza con una función f definida sobre los números reales x , la derivada de la función f' y una estimación inicial x_0 para una raíz de la función f . Si la función satisface las suposiciones hechas en la derivación de la fórmula y la estimación inicial está cerca, entonces una mejor aproximación x_1 es

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}$$

La idea del método es la siguiente: uno comienza con una conjetura inicial que está razonablemente cerca de la raíz verdadera, luego la función se aproxima por su línea tangente (que se puede calcular utilizando las herramientas de cálculo), y uno calcula la intersección de x de esta línea tangente (que se hace fácilmente con álgebra elemental). Esta intersección con X generalmente será una mejor aproximación a la raíz de la función que la suposición original, y el método puede ser iterado.

El método de Newton es una técnica extremadamente poderosa; en general, la convergencia es cuadrática: a medida que el método converge en la raíz, la diferencia entre la raíz y la aproximación es cuadrada (el número de dígitos exactos se duplica aproximadamente) en cada paso. Sin embargo, existen algunas dificultades con el método: dificultad para calcular la derivada de una función, falla del método para converger a la raíz, si no se cumplen los supuestos en la prueba de la convergencia cuadrática del método de Newton, la convergencia lenta para las raíces de multiplicidad mayor que 1.

1.3.2. Guía de usuarios – Guía del software: esta guía ayudará al usuario a moverse por el software y explicarle las diferentes ayudas, o complementos con los que cuenta el software, igualmente será de ayuda como esta guía especificando los componentes.

1.3.3. Guía de usuarios – Guía para la entrada de datos: En esta guía le ayudara al usuario para la entrada de datos para el buen funcionamiento del software ya que el software no reconoce ciertos caracteres matemáticos y deben convertirse para que el software la entienda, la guía seria la siguiente.

Guiar Para Entrada De Los Datos

Operadores Matematicos Para La Entrada Del Software

En cuanto a los operadores aritméticos, estos permiten realizar las diferentes operaciones aritméticas del álgebra: suma, resta, producto, división, ...Estos operadores de Python son de los más utilizados. El listado completo es el siguiente:

| OPERADOR | FUNCIÓN | COMO SE DEBE ESCRIBIR |
|------------|--------------------|-----------------------|
| + | suma | + |
| - | resta | - |
| X o * | multiplicación | * |
| ^ | potencia | ** |
| / | División(cociente) | / |
| % | División(resto) | % |
| Ln(x) | Logaritmo natural | Log(x) |
| Sen(x) | seno | Sin(x) |
| Cos(x) | coseno | Cos(x) |
| Tan(x) | tangente | Tan(x) |
| Sen^-1 | Arco seno | Asin(x) |
| Cos^-1 | Arco coseno | Acos(x) |
| Tan^-1 | Arco tangente | Atan(x) |
| Csc(x) | Cosecante | null |
| Sec(x) | Secante | null |
| Ctg(x) | cotangente | null |
| e | Euler | e |
| π | pi | pi |
| \sqrt{x} | raíz | Sqrt(x) |

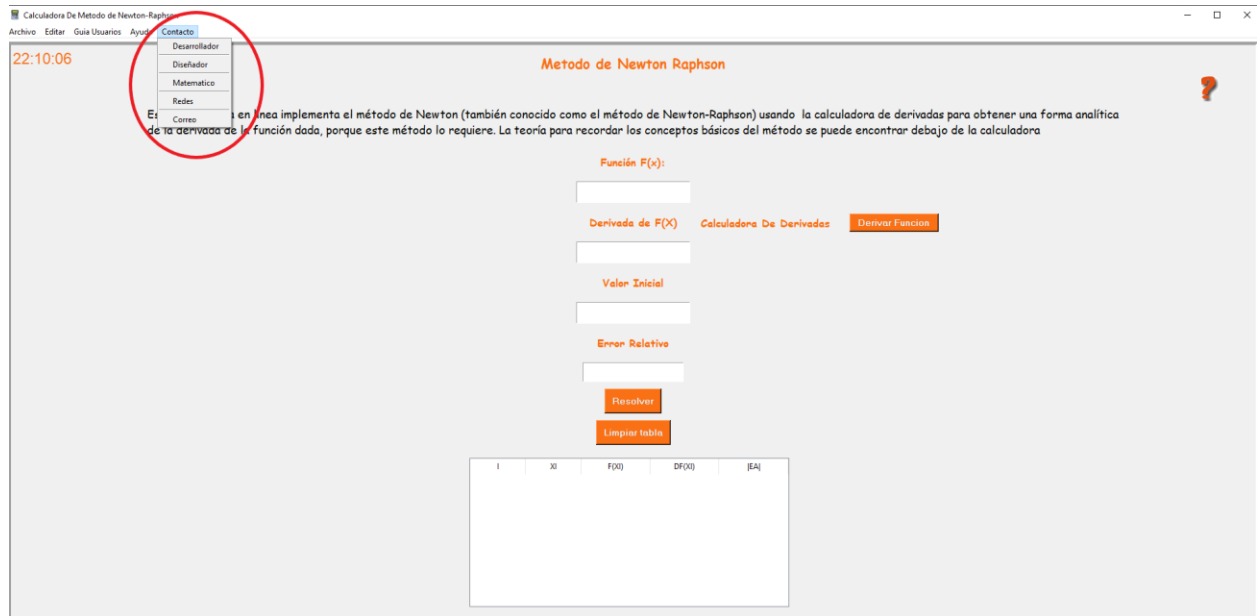
1.3.4. Guía de usuarios – Guía para el análisis de datos: En esta guía el usuario aprenderá a interpretar los diferentes resultados que el software le arrojará, mostrándole a que se refiere cada cosa, la guía seria la siguiente.

- 1.4.1. Ayudas – Ayuda:** en este apartado el usuario también tendrá la posibilidad de acceder a la guía de usuario, esto debido a que el usuario podrá ingresar desde diferentes partes a este manual tan importante
- 1.4.2. Ayudas – Acerca de:** en este submenú mostrara información acerca de la empresa o persona en este caso que desarrollo el software.



1.5. Menú del software – Contacto

En esta parte el software le posibilita al usuario los contactos de las diferentes personas que interactuaron en el desarrollo de dicho software.



1.5.1. Contacto – Desarrollador



1.5.2. Contacto – Diseñador



1.5.3. Contacto – Matemático



1.5.4. Contacto – Redes



1.5.5. Contacto – Correo

2. Cajas de Texto, para ingresar los datos

En estas cajas de texto que en el código se identificaría como los entry, son aquellas cajas de texto donde los usuarios deberán ingresar todos los datos necesarios para la utilización del software

2.1. Caja de texto – Función F(X): en esta caja de texto el usuario deberá ingresar las funciones para poder utilizar el método de newton Raphson. El usuario para poder llenar esta caja de texto donde ira la función, tendrá una **guía de entrada de datos** en la zona de **Guía de usuarios – Guía para la entrada de datos**, ya que el software no admite los caracteres comunes como (\wedge , $\text{sen}(x)$), deberán ser convertidos para que sea entendible al software.

2.3 Caja de texto – Valor inicial: en esta entrada el usuario podrá ingresar números enteros y números decimales (0 o 0.1 o 5 o 5.1) dando diversidad para los valores iniciales, el software no acepta caracteres como (. O , o * o ^,etc.), solo números.

The screenshot shows a web application titled "Calculadora De Metodo de Newton-Raphson". The main heading is "Metodo de Newton Raphson". Below the heading, there is a paragraph explaining the calculator's function. The interface includes several input fields and buttons:

- Función F(x):** A text input field.
- Derivada de F(X):** A text input field.
- Calculadora De Derivadas:** A button.
- Derivar Funcion:** A button.
- Valor Inicial:** A text input field, which is highlighted with a red circle.
- Resolver:** A button.
- Limpia tabla:** A button.

Below the buttons is a table with the following headers: **I**, **Xi**, **F(Xi)**, **DF(Xi)**, and **|Ea|**. The table is currently empty.

2.4 Caja de texto – Error Relativo: En el error relativo el usuario también podrá ingresar números enteros y decimales, tiene libertad de ingresar ese tipo de números, igualmente **no podrá ingresar caracteres que no sean numéricos**.

This screenshot shows the same web application as the previous one, but with the "Error Relativo" field highlighted with a red circle. The interface elements are identical to the previous screenshot, including the input fields for the function and its derivative, the "Valor Inicial" field, and the "Error Relativo" field. The table below the buttons remains empty.

3. Botones

En esta zona se explicará los requisitos, el funcionamiento y demás cosas de los botones que se encuentran en el software.

3.1. Botón – Resolver: requisitos para poder ejecutar este botón se deberá (llenar la caja de texto de "Función F(X)", llenar la caja de texto de "Derivada de F(X)", llenar la caja de texto de "Valor inicial", llenar la caja de texto de "Error relativo").

Luego de haber llenado las cajas de texto, cumpliendo los requisitos, podrá ejecutar el botón resolver.

El botón resolver ejecutara 2 tipos de funciones

1. Mostrar una ventana donde confirmara los datos ingresados
2. Resolver el método
3. Llenar la tabla con las iteraciones
4. Mostrar una respuesta oficial para el método
5. Borrar los campos de la caja de los textos.

Calculadora De Metodo de Newton-Raphson

Archivo Editar Guia Usuarios Ayuda Contacto

16:38:43

Metodo de Newton Raphson

Esta calculadora en línea implementa el método de Newton (también conocido como el método de Newton-Raphson) usando la calculadora de derivadas para obtener una forma analítica de la derivada de la función dada, porque este método lo requiere. La teoría para recordar los conceptos básicos del método se puede encontrar debajo de la calculadora

Función F(x):

Derivada de F(X) Calculadora De Derivadas Derivar Funcion

Valor Inicial

Error Relativo

Resolver

Limpiar Inputs

| i | x_i | $F(x_i)$ | $DF(x_i)$ | $ E_a $ |
|---|-------|----------|-----------|---------|
|---|-------|----------|-----------|---------|

3.2. Botón – Limpiar tabla: El botón limpiar tabla es un botón que el software ofrece para limpiar las tablas, ofreciendo una mejor comprensión de los datos en caso que el usuario quiera ingresar un nuevo ejercicio. El botón limpiar tabla realizara las siguientes funciones.

1. Limpiar la tabla donde se muestran las iteraciones y diferentes campos necesarios para la realización del método.
2. Limpiar la respuesta oficial que se encuentra debajo de la tabla.



4. Widgets del software

En este apartado el software ofrece 2 tipos de widgets los cuales uno es ilustrativo o informativo y un widget interactivo

4.1 Widgets – Widgets de pregunta: este widget con forma de interrogante el usuario al presionarle lo llevará a una ventana que le dará una guía de usuario menos detallada como la que se encuentra en este documento.

Conclusiones

El método de Newton es muy eficiente y rápido a la hora de encontrar raíces de una ecuación ya que presenta convergencia cuadrática, pero existen algunas excepciones en las cuales el comportamiento de la convergencia es totalmente diferente al cuadrático, esto sin tener en consideración algunos casos ya mencionados en los cuales el método necesita un análisis especial. Ahora bien, el problema de la convergencia del método ocurre cuando la ecuación tiene múltiples raíces y el procedimiento a seguir es modificar el algoritmo aplicando una nueva fórmula, garantizando así la eficacia y capacidad del método.

Bibliografía

METODO DE NEWTON RAPSHON

https://es.wikipedia.org/wiki/M%C3%A9todo_de_Newton

<https://www.geogebra.org/m/XCrwWHzy>

<http://personal.cimat.mx:8181/~julio/courses/progra01/clase18/MetododeNewtonRaphson.pdf>

http://www.ugr.es/~mpasadas/ftp/Tema2_apuntes.pdf

LENGUAJE DE PROGRAMACION PYTHON

<https://wiki.python.org/moin/BeginnersGuide>

<https://es.wikipedia.org/wiki/Python>

<https://www.crehana.com/blog/web/que-es-python/>

<https://www.oma.org.ar/invydoc/docs-libro/apuntes.pdf>

INTERFAZ GRAFICA TKINTER

<https://docs.python.org/3/library/tk.html>

<https://wiki.python.org/moin/TkInter>

LOGICA PYTHON

<https://www.freecodecamp.org/espanol/news/operadores-basicos-en-python-con-ejemplos/>

https://entrenamiento-python-basico.readthedocs.io/es/latest/leccion4/operadores_logicos.html

<https://docs.hektorprofe.net/python/operadores-y-expresiones/operadores-logicos/>

