Numerical Analysis User manual

Maria Sofía Uribe,Santiago Tello,Luis Herrera Chamat,Andrés de Jesús Martinez

November 20, 2020

Contents

1 Repository Address

https://github.com/msuribec/AnalisisNumerico

2 Members

- María Sofía Uribe
- Santiago Tello
- Luis Herrera Chamat
- Andrés de Jesús Martinez

3 User Manual

3.1 Requirements to run the program

The interactive graphical user interface for the app was developed in Python 3. It is necessary that python 3 is installed since the division operator behaves differently in previous versions of Python.

Some aditional libraries needed are:

- tkinter (some distributions of Python 3 will have this already installed depending on the Operating system that is being used)
- matplotlib
- sympy
- numpy

3.2 Interacting with the main window

The main menu of the app can be accessed by running the **NumericalAnalysisGUI.py** module in the python project folder.

This will prompt the main menu.The main menu has 6 buttons, one for each type of method. The user must choose the type of method they want to try.

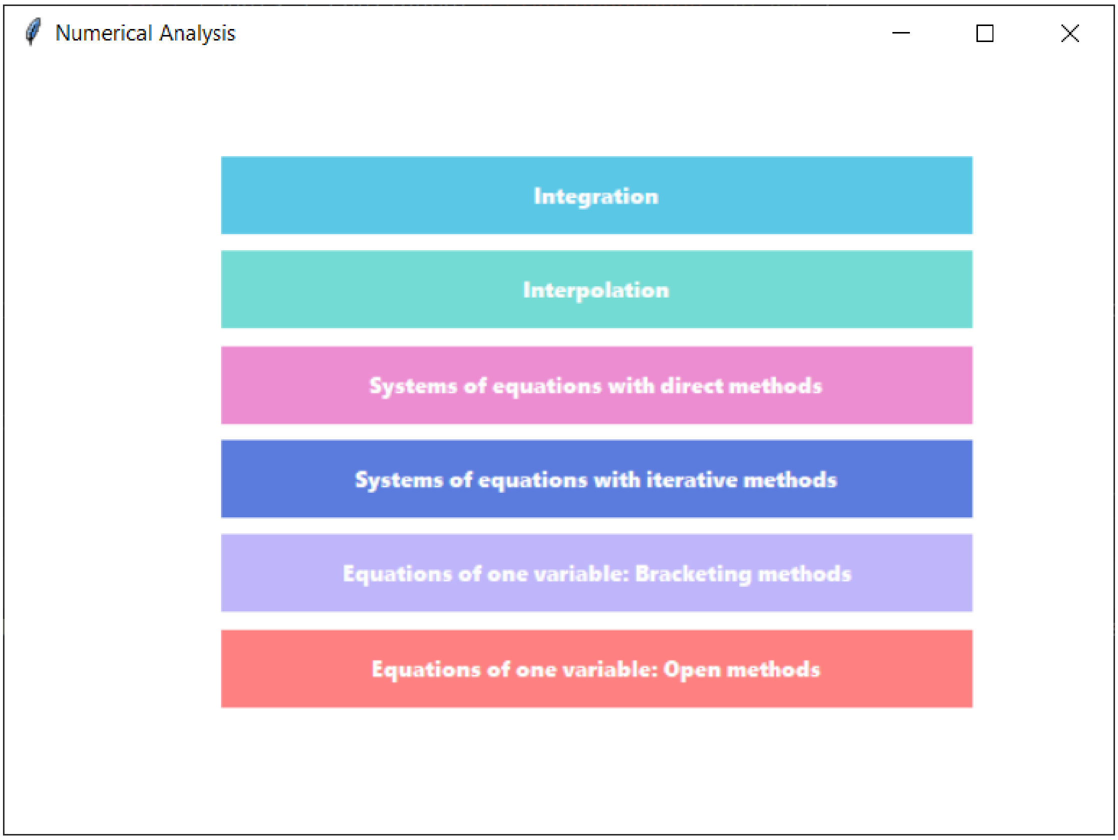

Figure 1: Main menu

3.3 Integration methods

Clicking the Integration button on the main menu will prompt the window for the integration methods.The user must choose the name of the method they wish to run.The options are trapezoid method, generalized trapezoid method,simpson 1/3 method, generalized simpson 1/3 and simpson 3/8.

For each method the user must input a function and the limits of integration (a is the lower limit and b is the upper limit)

If the chosen method is one of the generalized methods, the user must choose the number of intervals (must be an integer and in the case of generalized simpson 1/3 it must be even).

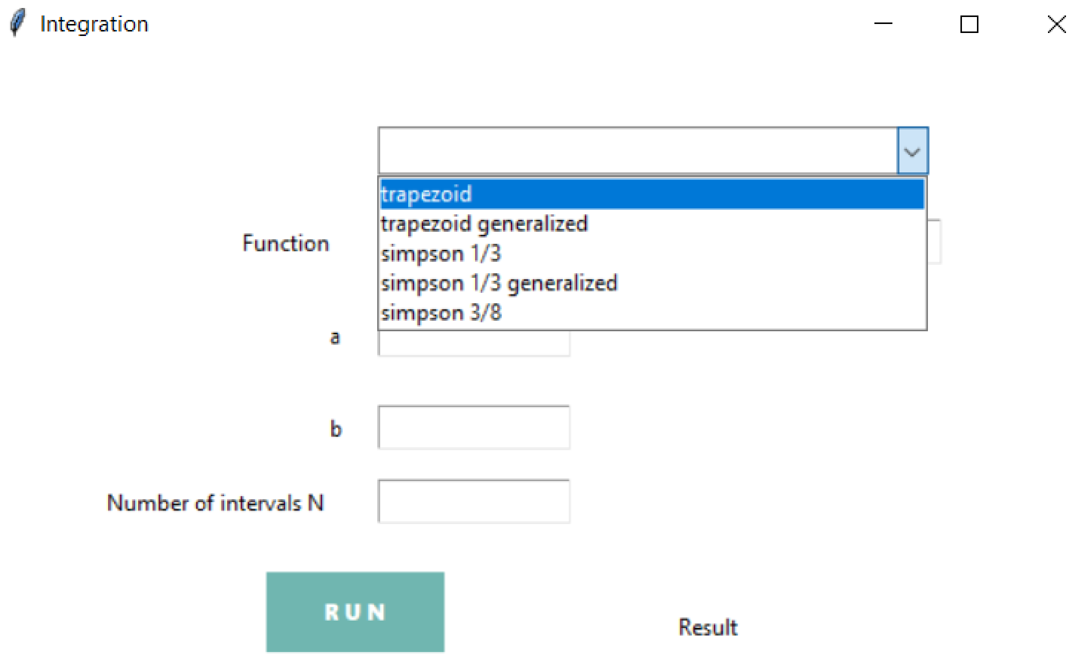

Figure 2: Window for integration methods

Examples of functions are

- e^x is written as exp(x)
- $x^2 - 4$ is written as x**2 -4

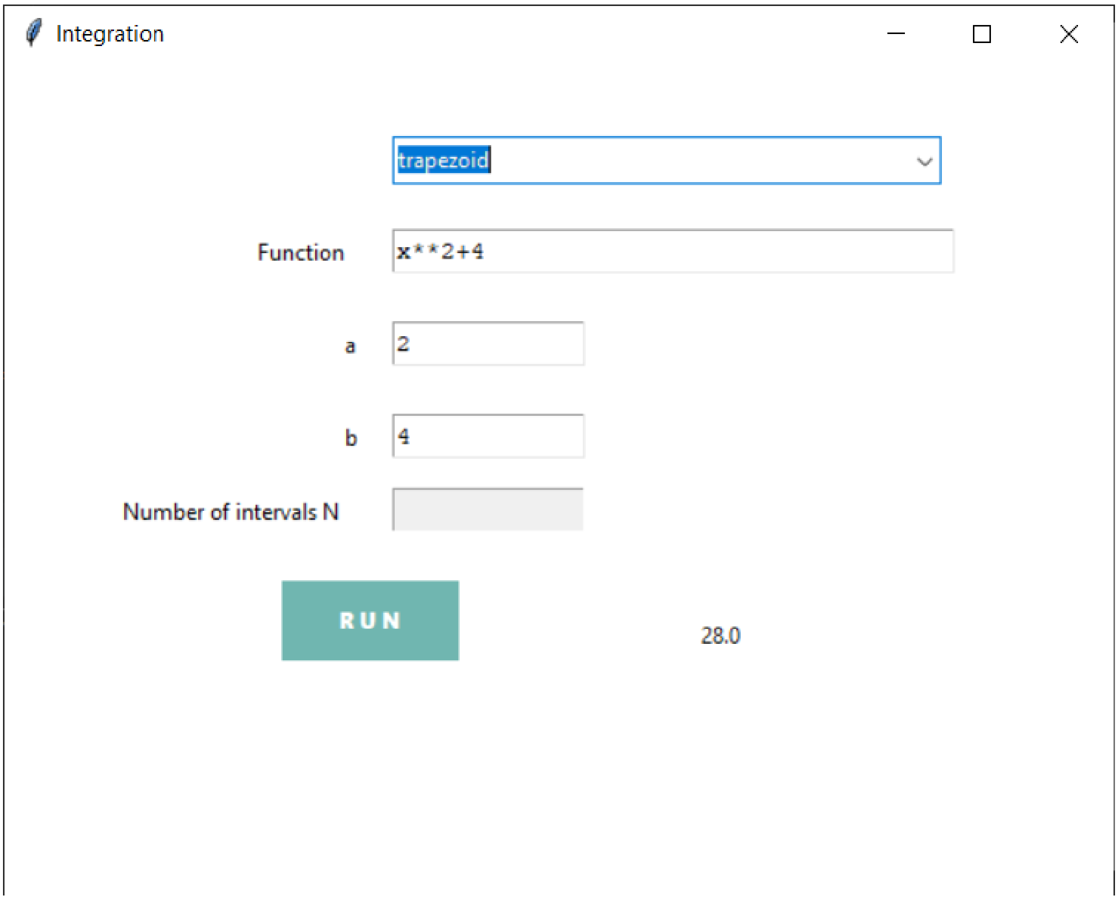

Figure 3: Example of input for the trapezoid method, the function is $f(x) = x^2 - 4$

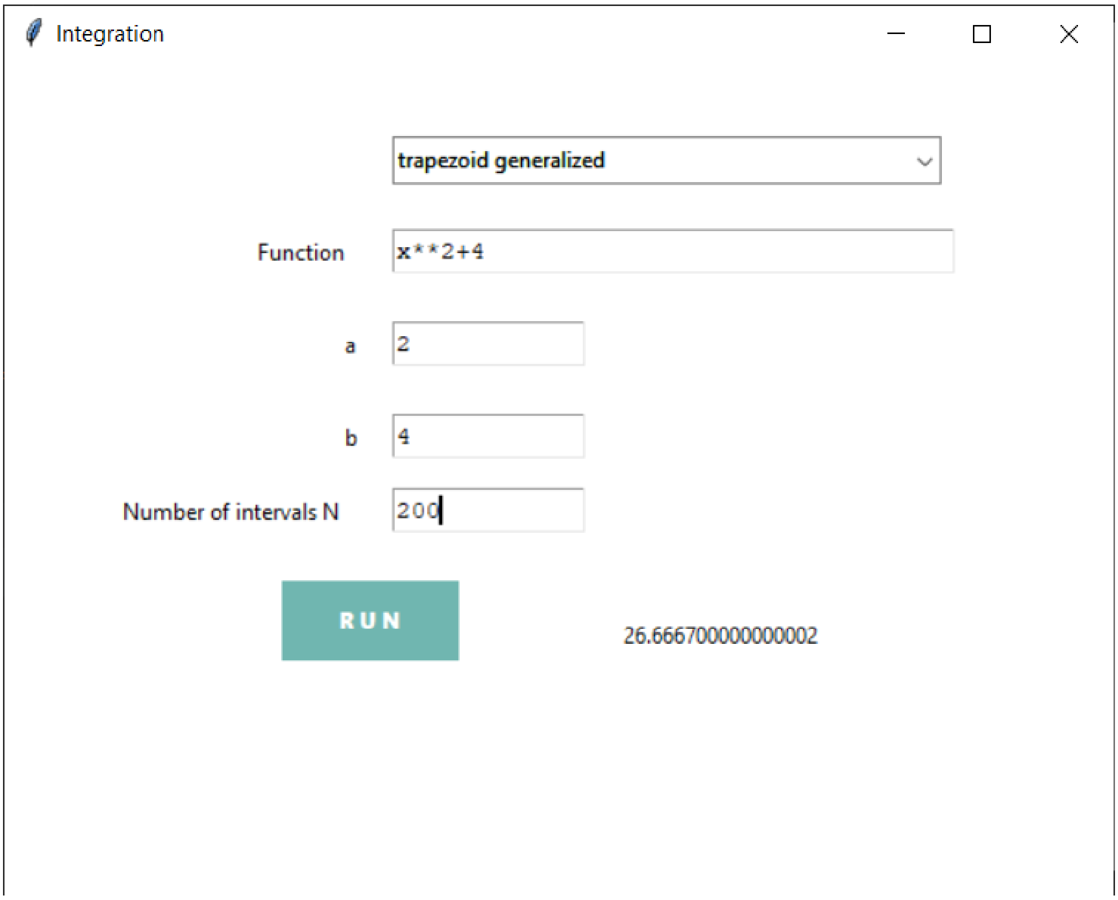

Figure 4: Example of input for the generalized trapezoid method, the function is $f(x) = x^2 - 4$

3.4 Interpolation

Clicking the Interpolation button on the main menu will prompt the window for the interpolation methods.The user must choose the name of the method they wish to run.The options are the Vandermonde method, the divided differences method and the Lagrange method.

For each method the user must input the x values and the y values separated by a comma. e

For example, to interpolate the points $(1,1),(2,4),(3,9)$ the x values must be entered as $1,2,3$ and the y values as $1,4,9$

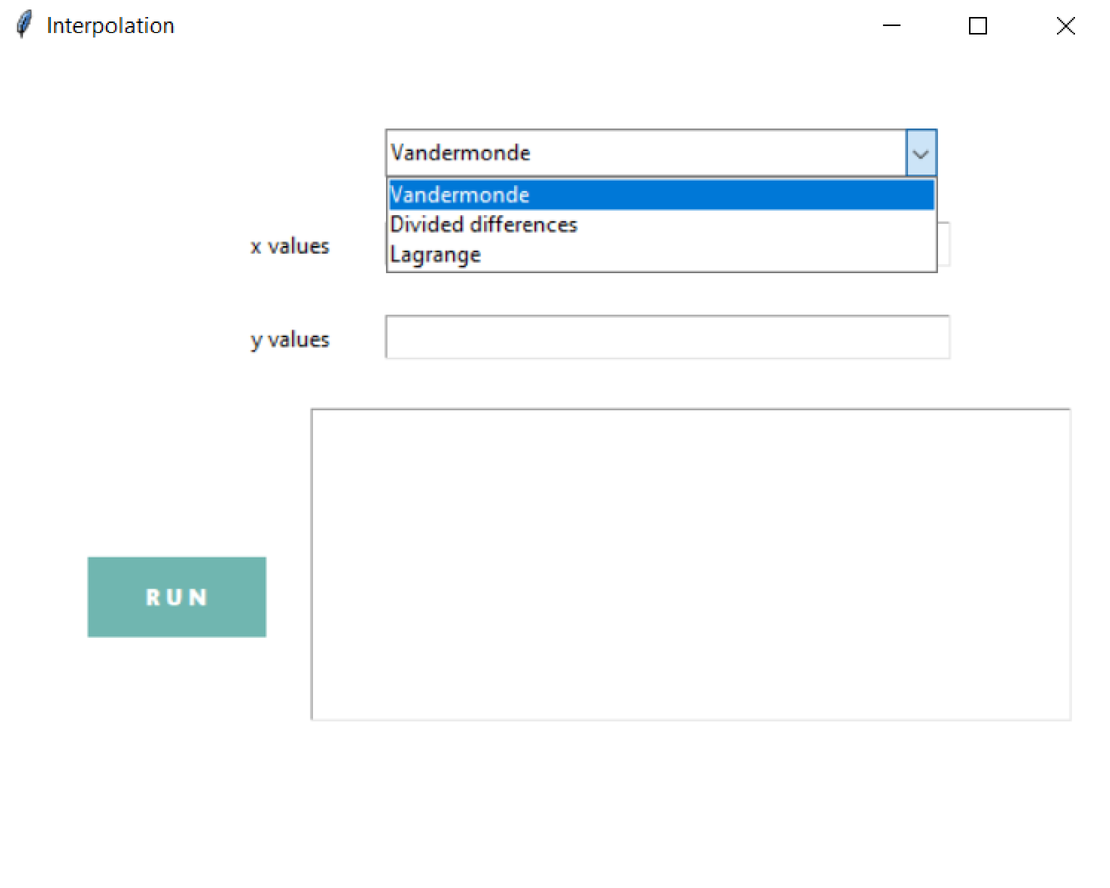

Figure 5: Window for interpolation methods

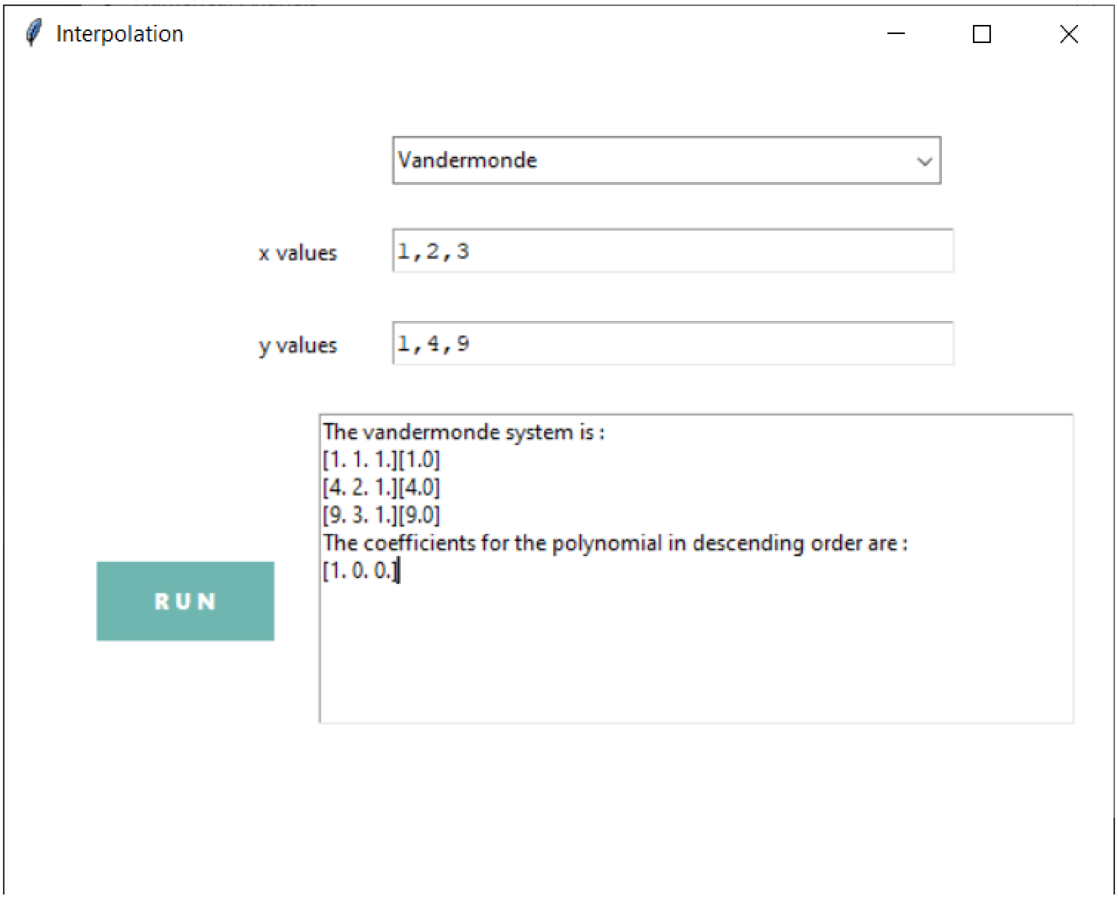

Figure 6: Example of input for the Vandermonde method

3.5 Systems of equations

3.5.1 Direct methods

Clicking the Systems of equations with direct methods button on the main menu will prompt the window for the Systems of equations with direct methods. The user must choose the name of the method they wish to run.The options are Gaussian elimination, Gaussian elimination with partial pivoting, Gaussian elimination with total pivoting, LU factorization with Gaussian elimination, LU factorization with Gaussian elimination and partial pivoting, Cholesky factorization,Crout factorization and Doolittle factorization

For each method the user must input the coefficient matrix A(all elements separated by a comma) and the vector of independent terms b to solve the system $Ax = b$ (all elements separated by a comma)

For example, the matrix

$$A = \begin{bmatrix} 129 & 163 & 70 \\ 195 & 142 & 156 \\ 357 & 300 & 276 \end{bmatrix}$$

must be entered as

$$129, 163, 70, 195, 142, 156, 357, 300, 276$$

and the vector

$$b = \begin{bmatrix} 10 \\ 33 \\ 43 \end{bmatrix}$$

must be entered as

$$10, 33, 43$$

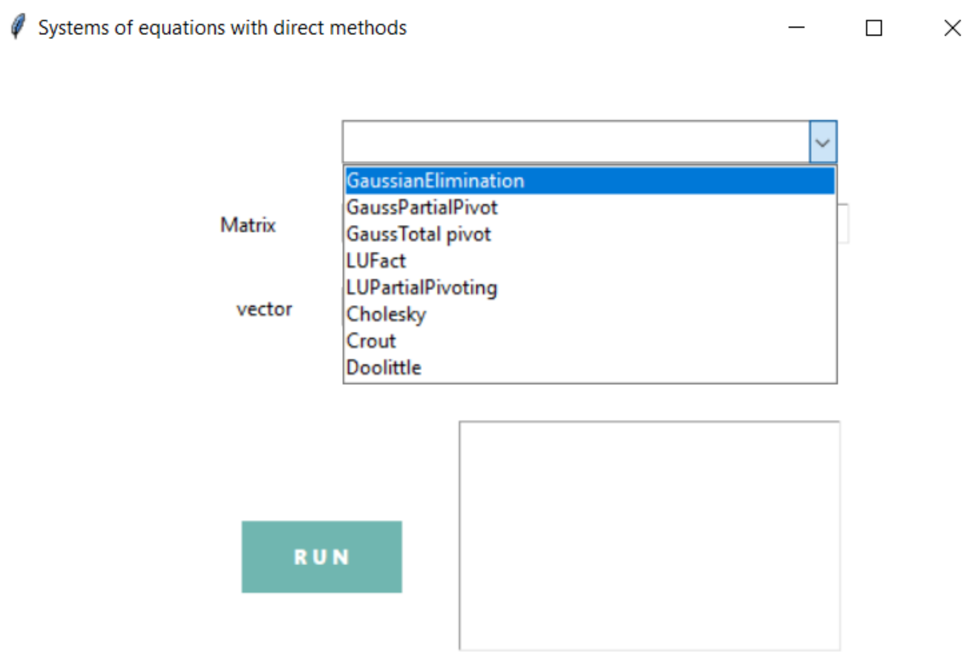

Figure 7: Window for Systems of equations with direct methods

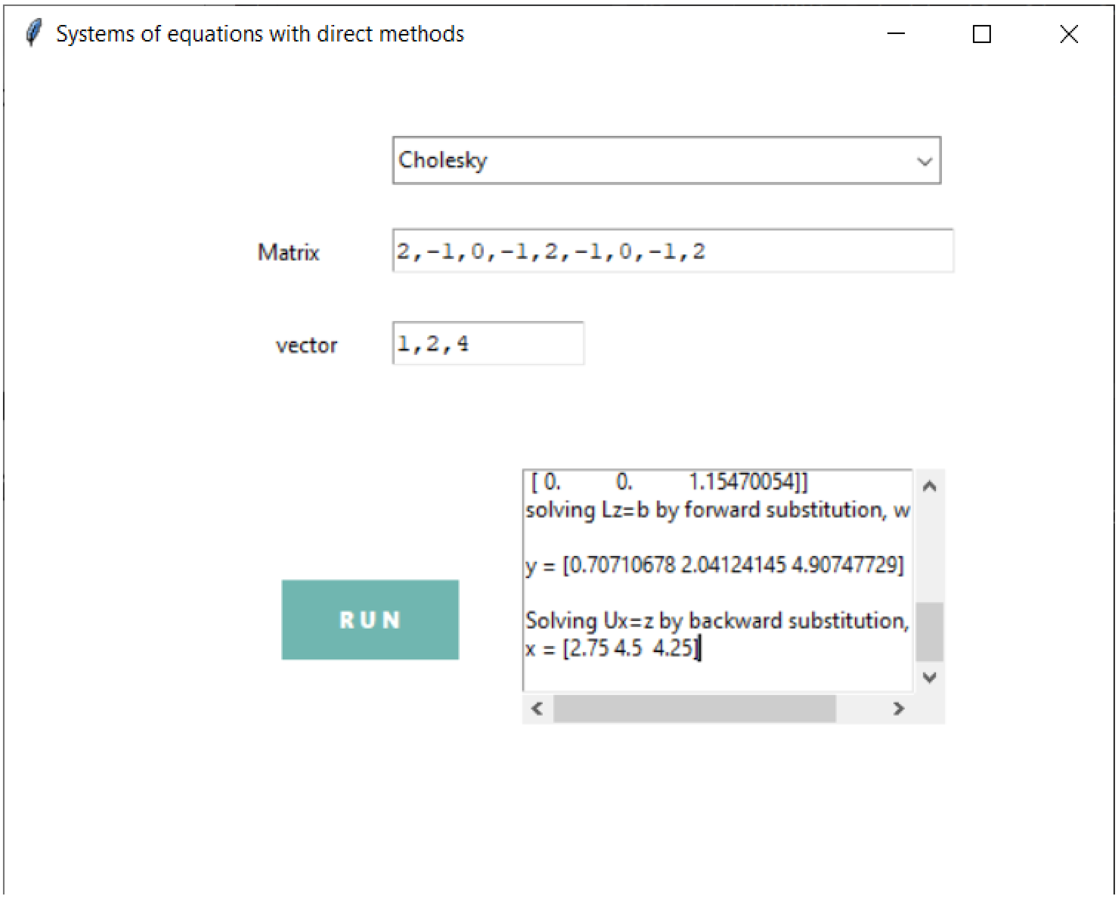

Figure 8: Example of input for the Cholesky method

3.5.2 Iterative methods

Clicking the Systems of equations with iterative methods button on the main menu will prompt the window for the Systems of equations with iterative methods. The user must choose the name of the method they wish to run.The options are Jacobi and Gauss Seidel.

For each method the user must input the coefficient matrix A(all elements separated by a comma) and the vector of independent terms b to solve the system (all elements separated by a comma)$Ax = b$. Additionally the user must enter the number of maximum iterations (must be an integer) and the tolerance(float with dot as a decimal separator).

For example, the matrix

$$A = \begin{bmatrix} 129 & 163 & 70 \\ 195 & 142 & 156 \\ 357 & 300 & 276 \end{bmatrix}$$

must be entered as

$$129, 163, 70, 195, 142, 156, 357, 300, 276$$

and the vector

$$b = \begin{bmatrix} 10 \\ 33 \\ 43 \end{bmatrix}$$

must be entered as

$$10, 33, 43$$

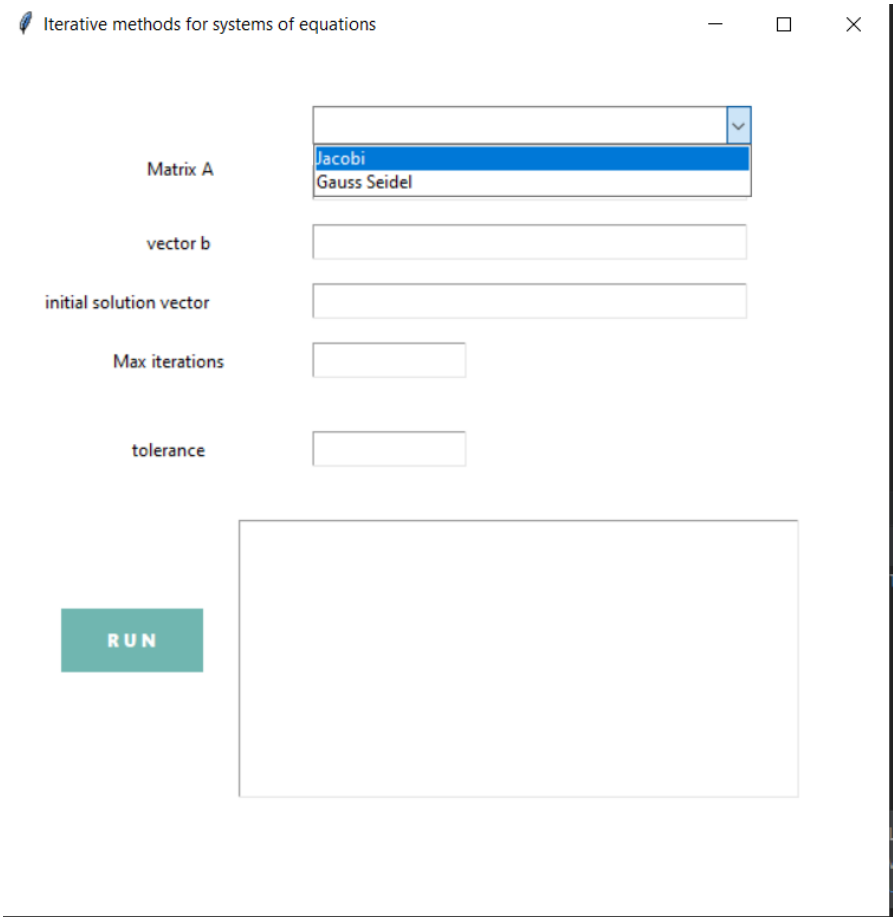

Figure 9: Window for Systems of equations with iterative methods

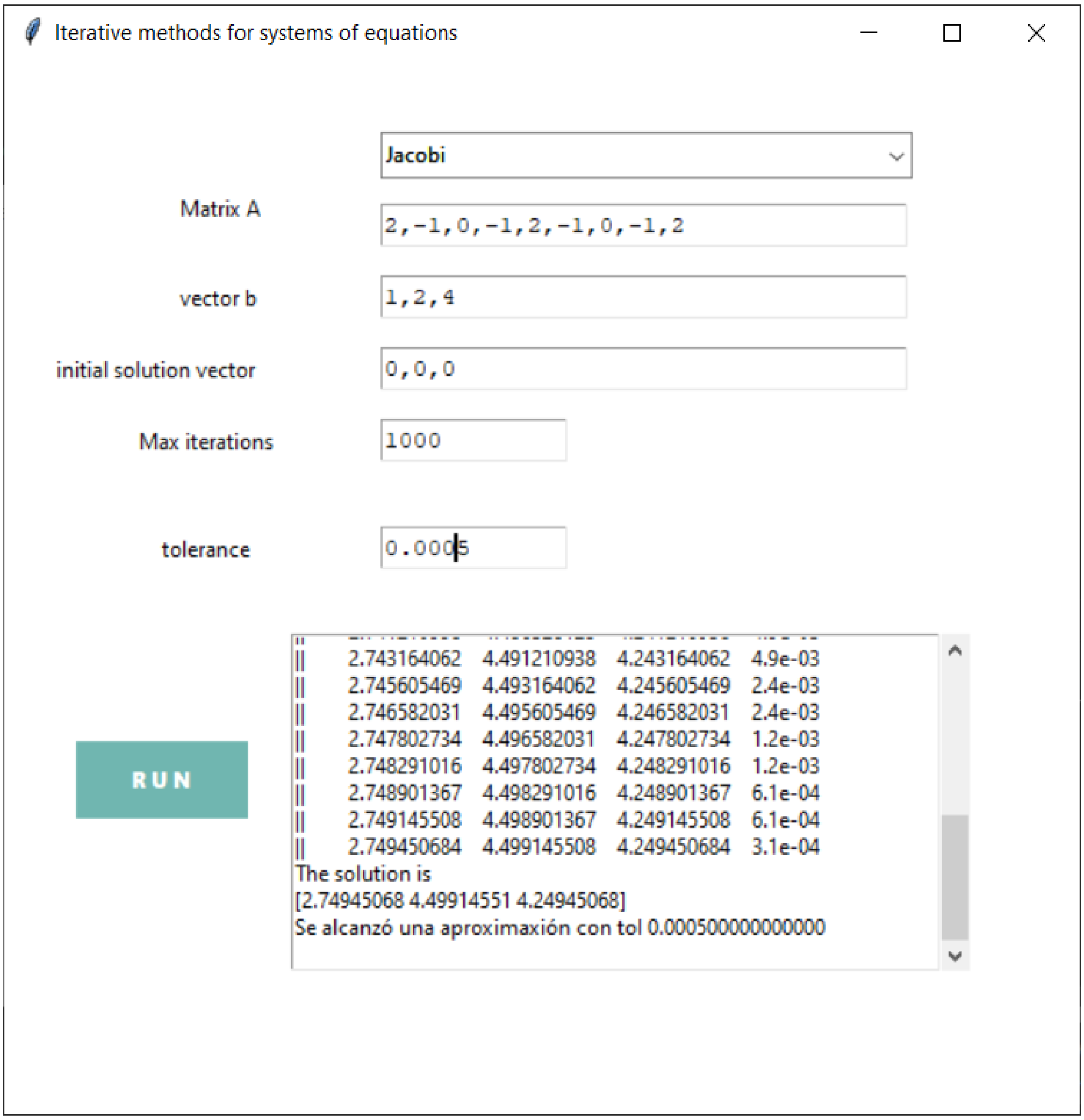

Figure 10: Example of input for the jacobi method

3.6 Equations of one variable

Clicking the Equations of one variable equations with bracketing methods button on the main menu will prompt the window for the one variable equations with bracketing methods.The user must choose the name of the method they wish to run.The options are bisection and regula falsi.

For each method the user must input a function and the values of a and b , the number of maximum iterations (must be an integer) and the tolerance(float with dot as a decimal separator).

Examples of functions are

- e^x is written as exp(x)
- $x^2 - 4$ is written as x**2 -4

3.6.1 Bracketing methods

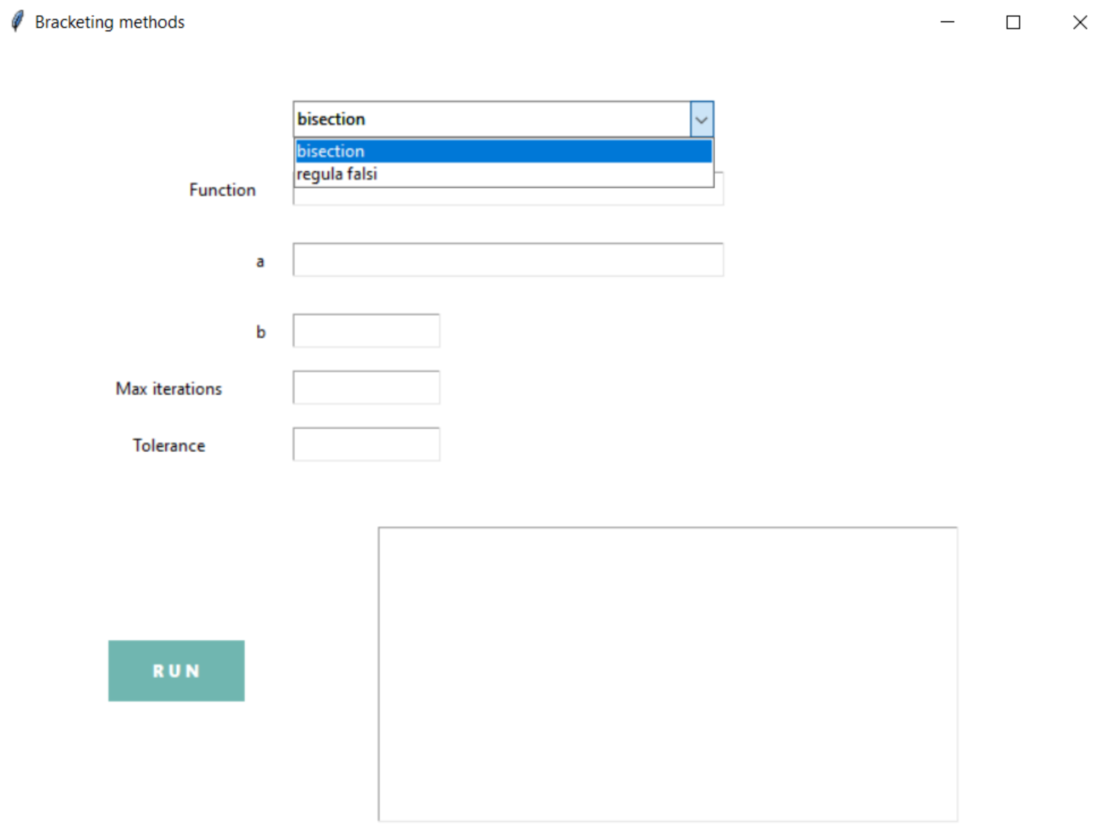

Figure 11: Window for bracketing methods

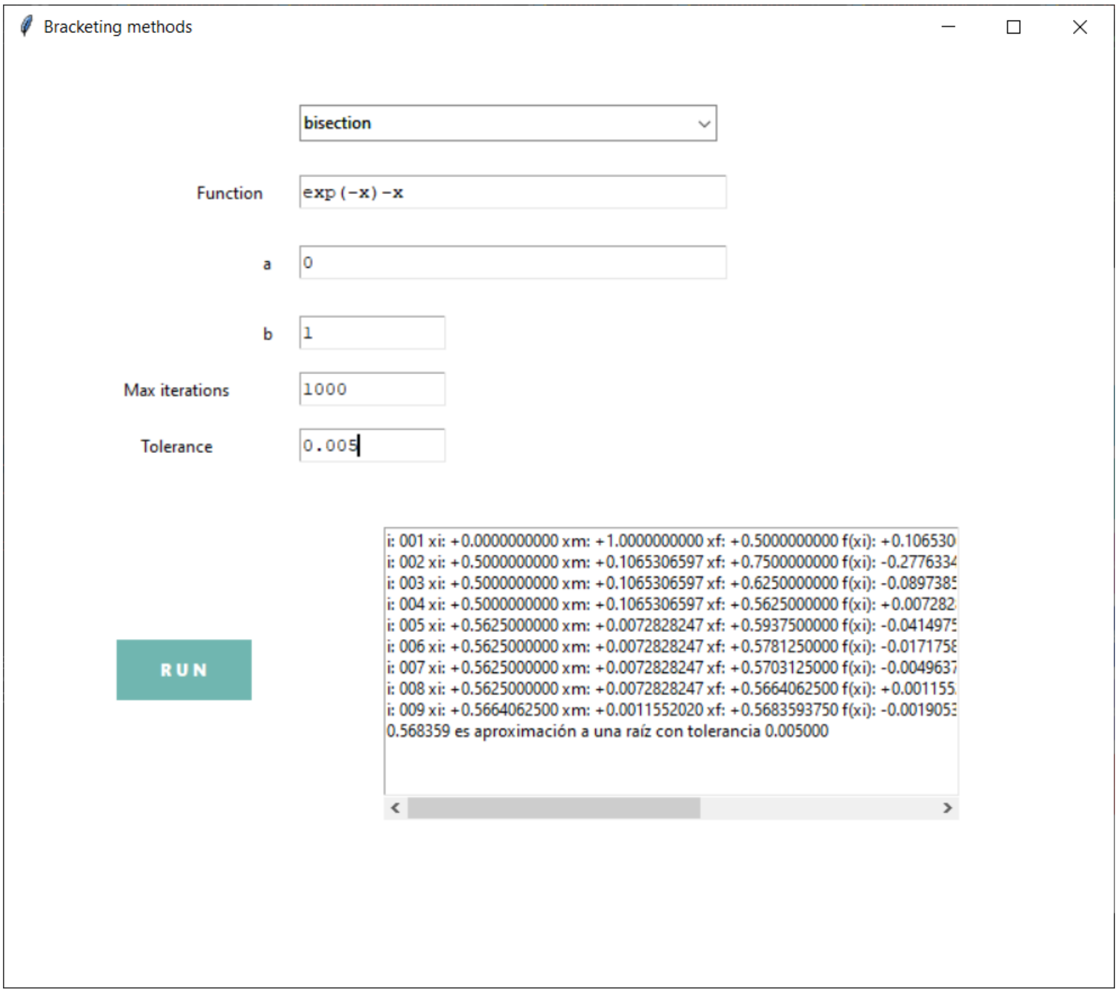

Figure 12: Example of input for bracketing methods

3.6.2 Open methods

Clicking the Equations of one variable equation with open methods button on the main menu will prompt the window for the one variable equation with open methods.The user must choose the name of the method they wish to run.The options are Fixed point,Newton,Secant and multiple roots.Additionally, there is an option to run the incremental searches method (this is not an open method for root finding but is included in case the user wants to use it to find an initial approximation for the other methods)

For each method the user must input a function f, the value of the initial approximation,x_0,the number of maximum iterations (must be an integer) and the tolerance(float with dot as a decimal separator).

For the fixed point method, there is an additional function g that is needed. For the secant method an additional initial value x_1, is needed.

These fields will only be available when the respective methods are chosen, otherwise they will be disabled to avoid confusion

Examples of functions are

- e^x is written as exp(x)
- $x^2 - 4$ is written as x**2 -4

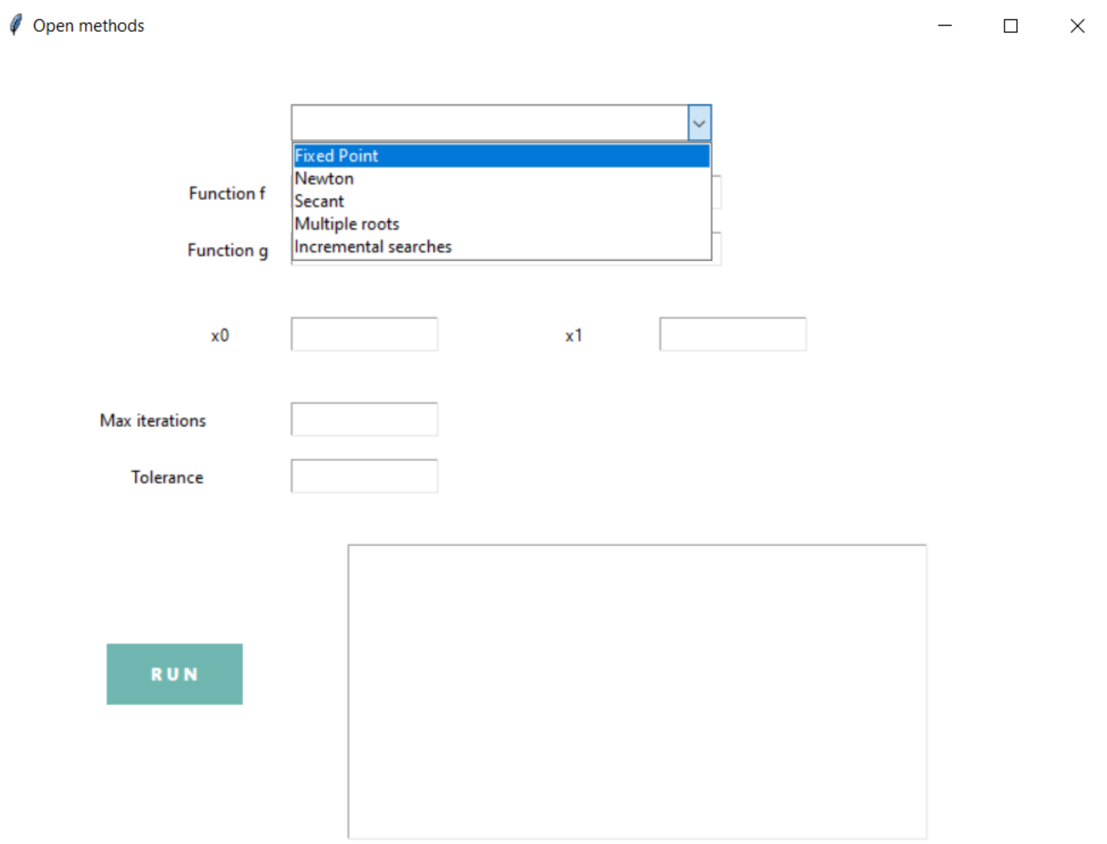

Figure 13: Window for equations of one variable with open methods

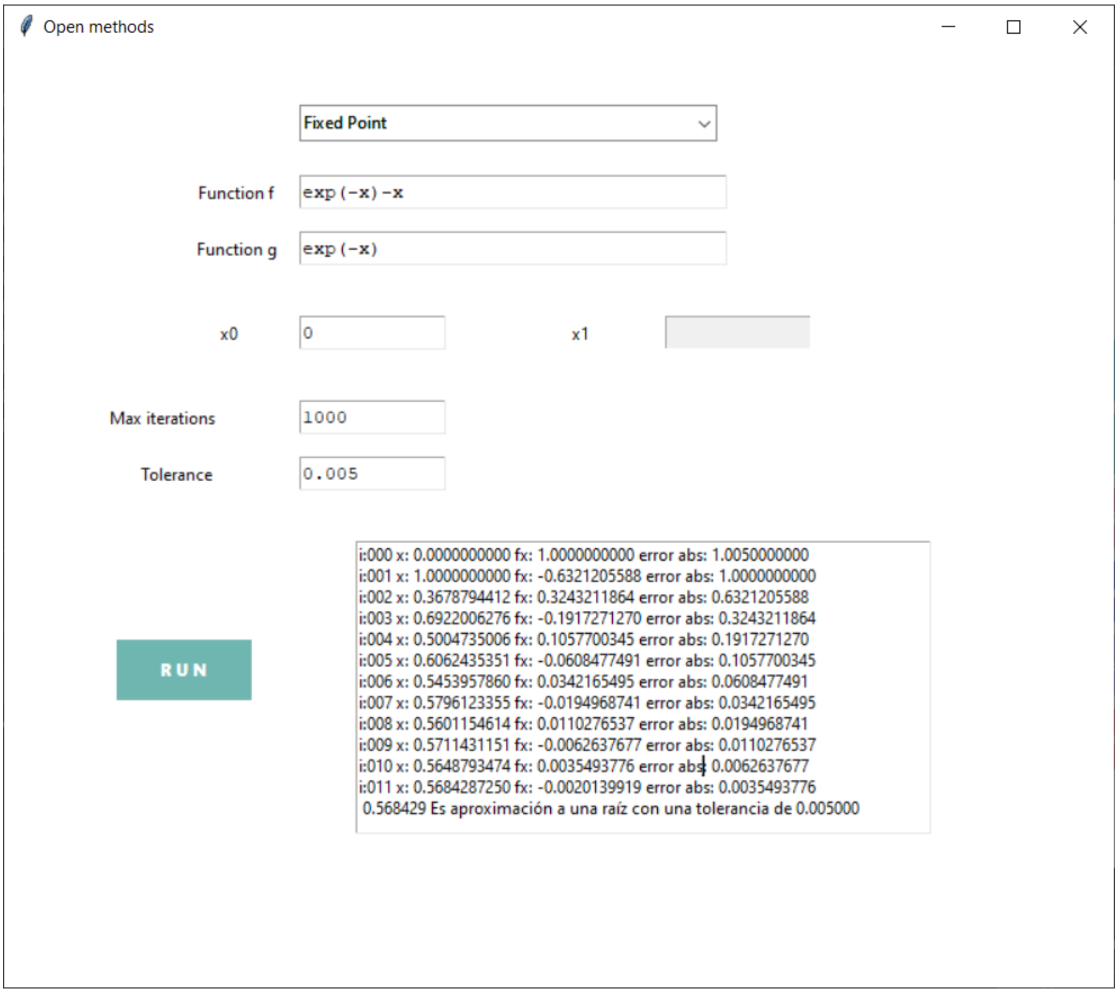

Figure 14: Example of input for the fixed point method

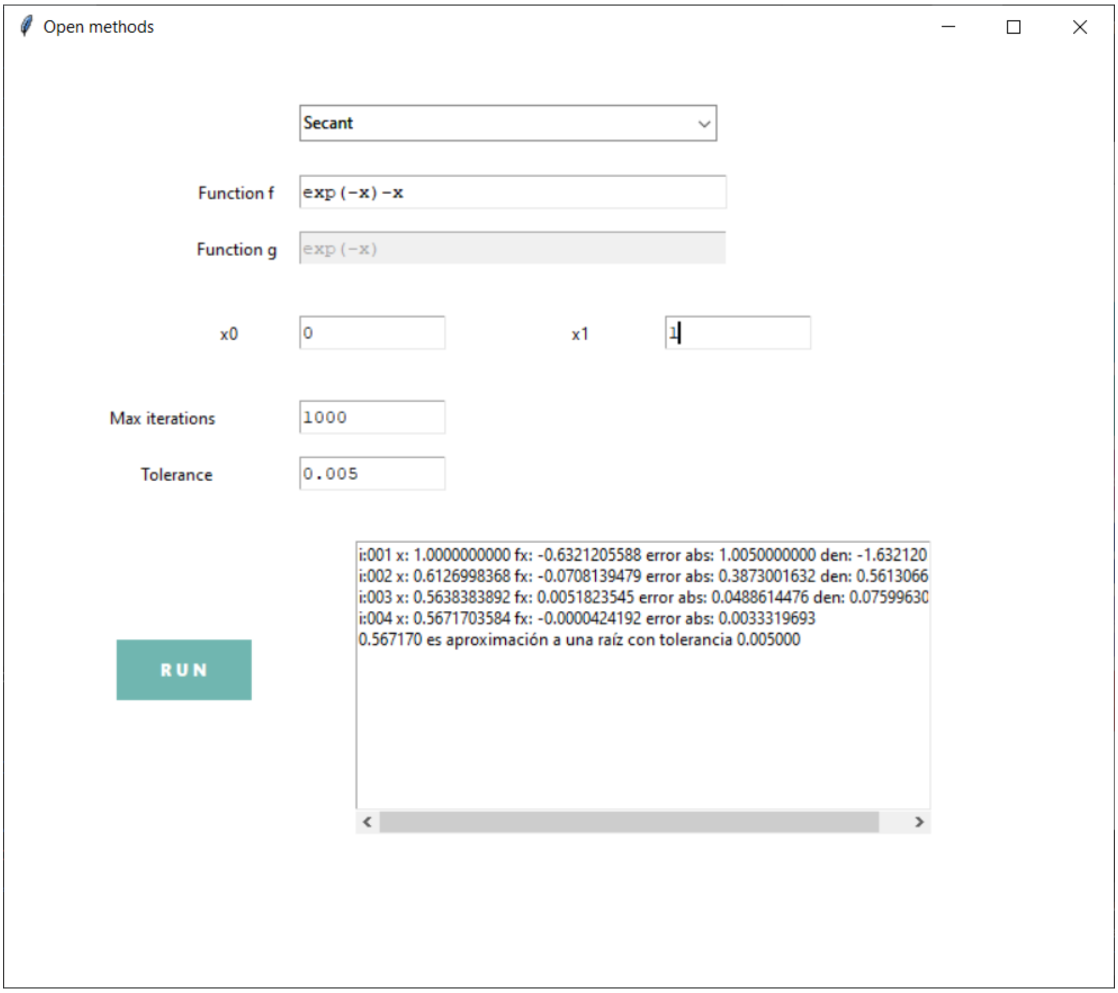

Figure 15: Example of input for the secant method