Project 2: Runge-Kutta-Fehlberg (RKF) for ODE

Austin Johns

Grand Canyon University

CST-305: Principles of Modeling and Simulation

**Installing Required Software/Libraries**

Per the needs of this project, we imported math, matplotlib.pyplot as plt, and time. The software used to execute the program was PyCharm.

A close up of a logo

Description automatically generated

**Execution Without Error**

The execution of the program yields many data points that are plotted to a graph. Using the Runge-Kutta method and a recursive algorithm, the program plots 1000 data points to find

A screenshot of a social media post

Description automatically generatedx1-1000, y1-1000.

A picture containing computer

Description automatically generated

**Mathematical Methods in Code**

In the code, the first step of carrying out the Runge-Kutta method would be to find k1-k4. This was executed by assigning the functions of each k value and computing the calculations.

A close up of a screen

Description automatically generated

From there, the T4 value is calculated using the k values.



After this, these values are used to find the new x and y values.



Next, a loop is included in the code to repeat this process to (x1000, y1000).

A screenshot of a cell phone

Description automatically generated

**Output Production**

The output for this project is fairly simple. The output prints each x and y value, 1-1000, how many steps of computation there was to get there, and the time it took to compute the entire process. There’s also a plot made for the graph that compares the x and y values.

A picture containing meter, clock

Description automatically generated

**Programming Style**

In this project, the programming style is straightforward. The naming convention for the variables in the project was given because of the x, y, and k values that were used. A couple of equations and calculations were used when defining variables to save space and make the programming style as efficient as possible. The code has been commented and sectioned into steps.

**Runge-Kutta Process**

The Runge-Kutta process is a tool used to solve differential equations. This process begins with a differential equation and uses defined variables for x, y, and h. These variables are used in a four-step process to find k values that are used to find a T4 value. The T4 value is used to find the solved y value. In k4, the x value is regarded as the new solved y value.

**Mathematical Method**

The mathematical method can be seen below, as the first four x and y values were calculated by hand.

A screenshot of a cell phone

Description automatically generated A screenshot of a cell phone screen with text

Description automatically generated A close up of text on a white background

Description automatically generated A screenshot of a cell phone

Description automatically generated A screenshot of a cell phone

Description automatically generated

**Mathematical Accuracy**

The accuracy of the hand-done calculations and the calculations of the program are 100%. As seen below, there is a screenshot of the first four x and y values of the program compared with the results of the hand-done calculations.

A close up of a sign

Description automatically generated

A close up of a logo

Description automatically generated

**Execution Screenshots**

**A screenshot of a cell phone

Description automatically generated**

**README File Screenshot**

**A screenshot of a cell phone

Description automatically generated**