How does the euler.m function work?

1 About Functions

For this assignment, you need to modify the euler.m m-file. You are given a template and will need to add a few lines of code so that euler.m will calculate the solution of an initial value problem using Euler's method. **Do not modify any of the code in the template.**
Let's take a look at the *euler* function declaration: **function** [t y]=euler(file_name, tvalues, y0, N).

1. **[t y]** is the output of the function. t and y are output variables. In this assignment they represent vectors. For now, we can think of a vector as an ordered list of (one or more) values.
2. **euler** is the name of the function.
3. **file_name, tvalues, y0, N** are labels that represent the input values of the function. When someone wants to run the *euler* function, he must give it some input values. We can use and manipulate these labels in the body of our function. It's important to realize that **the labels do not mean anything intrinsically**. Good programmers choose names for the labels that make sense; for instance, we could use A,B,C,D for our input labels, but this would make our function difficult to understand.

2 Adding Code

We can find the Euler algorithm on page 35 of our text. Before we add any code, we ought to look at the inputs that the user will give us, since we will need to use these somewhere in our function. Our first job is to figure out what information do we need for the algorithm and where will we get that information. Once we have done that, we can work on:

•Setting up the initial conditions.
•Setting up the iterations.

Setting up the initial conditions

It makes sense to tackle the initial conditions first. Since [t y] will be the output of our (t,y) values, we need to consider:

- what should be our first value of t?
- what should be our first value of y?
- how do we calculate h?

Setting up the iterations

In the loop, notice that we are starting at i=2. To see why the starting value of i might be relevant, we should look at the incremental part of the Euler method.

$$t_{n+1} = t_n + h \tag{1}$$

$$y_{n+1} = y_n + h * f(t_n, y_n) \tag{2}$$

We will be calculating the next value of t and y from the current value of t and y. So, for example, $t_2 = t_1 + h$ and $y_2 = y_1 + hf(t_1, y_1)$. Since i begins at 2 we will need to index our t() and y() accordingly. The loop will run as long as the value of i is less than or equal to N+1. Then the function will end. You will need to take this pseudocode (equations 1 & 2) and translate it into Matlab code.

3 Calling the Function

We can call the *euler* function from the command line in a way similar to calling *ode45*. We type euler('diff_eq', [0 1], 1, 5) for instance. This will pass the required input to *euler*. It will then solve the initial value problem in diff_eq.m as t goes from 0 to 1 for y(0)=1 with a step-size of $\frac{1-0}{5} = 0.2$.

Help

If you have any questions about some of the provided code, first take a look through Matlab's help file. If you have any remaining questions, please feel free to ask.