Lab 7: Implicit Euler

Background

A general equation for the growth rate of cells can be expressed as:

$$\frac{\mathrm{d}X}{\mathrm{d}t} = \mu X - k_d X^2$$

where X is cell concentration, μ is the cellular growth rate, and k_d is a death rate that incorporates the effects of cellular competition for limited resources. This differential equation serves as a practical example for solving differential equations.

Task

Use an implicit Euler method to trace cell concentration (in cells per mL) over time given an initial cell concentration and time step size (in hours). You may take the equation constants to be μ = 1e-1 and k_d = 1e-6 (you may also assume that these constants are appropriate for all of the stated units). Recall that solving an implicit Euler equation requires solving a non-linear equation. You may do this analytically or using a non-linear solution method of your choice (such as the Newton method).

Core requirements

Function name	`cell_growth`
Input	Initial cell concentration (in cells per mL), time step size (in hours), maximum time (in hours)
Output	A vector of values corresponding to the cell concentration at each time step
Example	`cell_growth(1e4, 1, 5)` should give `[10000, 10977, 12036, 13180, 14414, 15740]`

Bonus requirements

Generalize the given equation by allowing a fourth argument as a vector of length two specifying the μ and k_d terms. Note, your function should still work without a fourth argument — you will need to look up how to use the `varargin` term.

Function name	`cell_growth`
Input	Initial cell concentration (in cells per mL), time step size (in hours), maximum time (in hours), a vector of two values specifying μ and k_d
Output	A vector of values corresponding to the cell concentration at each time step
Example	`cell_growth(1e4, 1, 5, [2e-1, 1e-7])` should give `[10000, 12481, 15570, 19416, 24197, 30132]`

Submission checklist

Use the following checklist to help ensure you are meeting the core lab requirements:

- ☐ You are submitting a single file called `cell_growth.m` which defines a function called `cell_growth`
- ☐ Your function returns a value:
  `cell_growth(1e4, 1, 5)/2` should not cause problems
- ☐ Your function should output the right number of values depending on the time step:
  `cell_growth(1e4, 2, 5)` should output 3 values: one for 0 h, 2h, and 4h