

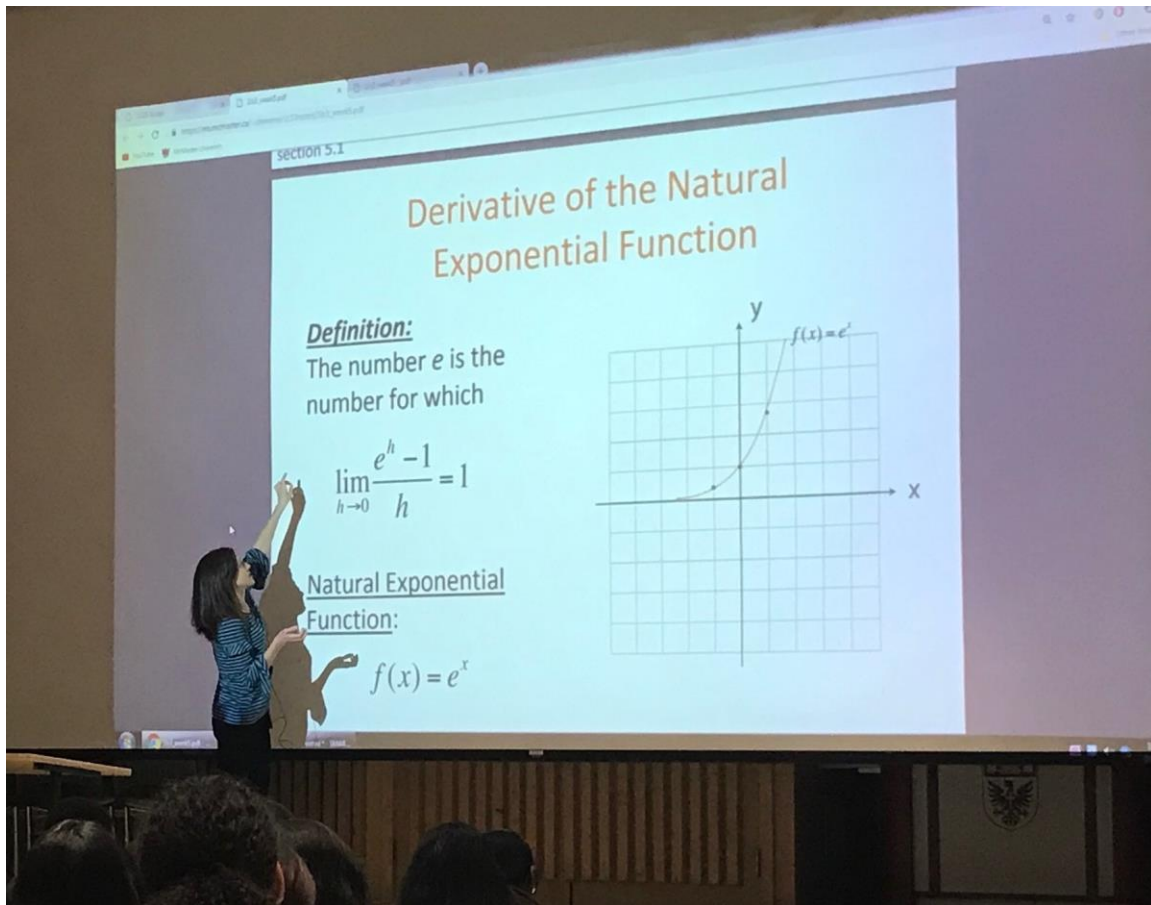
Computational Thinking in Tertiary Mathematics Education

How does computational thinking affect students' understanding and experiences in tertiary mathematics?

By Erin Clements, McMaster University, October 2018

Background:

I have been teaching the undergraduate course Calculus for Life Sciences (Math 1LS3) for approximately 10 years at McMaster University. Each semester I watch students struggle with applying mathematics in context, that is, connecting the abstract concepts taught in a traditional mathematics course to the practical mathematics found in journal articles.



Through my work as a research assistant for Dr. George Gadanidis, I was introduced to several math and coding initiatives being implemented by the Wellington District Catholic School Board. What I found most intriguing was a dual credit course, which integrated mathematics and computer science, thus covering two streams of material at once. This integrated approach was appealing to me for a number of reasons and it inspired me to research a similar approach at the tertiary level.



Why integrate coding into an applied calculus course?

Incorporating coding activities into Math 1LS3 provides an alternative way to explore the mathematical concepts traditionally taught in a first-year calculus course. This has the potential to increase the engagement of students (and instructors!), while better accommodating a diverse group of students with varying learning styles and abilities. As well, exploring mathematical models and algorithms using code offers a more dynamic experience for students, which traditional paper and pencil activities cannot provide. For example, students are able to make predictions, reason intuitively about relationships, and easily interact with the variables and parameters and receive immediate feedback. Moreover, integrating technology into our mathematical explorations allows us to keep the applications meaningful and authentic (for instance, we are not forced to oversimplify examples to make them more manageable to compute), and better aligns our activities with current practices in the field. By integrating this component into a preexisting mathematics course, students have an opportunity to enrich their mathematical understanding and develop a working knowledge of a modern computer language (a valuable skill in an increasingly computational world!), without taking an additional course explicitly in computer programming.

An example:

Euler's method is an algorithm used to approximate solutions of first-order differential equations, given initial conditions. In Math 1LS3, students are asked to approximate the solution to a single differential equation by applying Euler's method for a maximum of four steps, thus obtaining a superficial, uninteresting, yet easy to compute, result.

7. The change in the number of people infected with the Ebola virus in Liberia in 2014 has been modelled by the initial value problem

$$I'(t) = 10.5\sqrt{t} + 2e^{-0.1t}, \quad I(0) = 26$$

where time t is measured in days, and $t = 0$ corresponds to August 1st, 2014.

(a) [2] Estimate the number of individuals infected by August 5th, 2014 using Euler's method and a step size of 2 days. Round your answer to the nearest integer.

$$\text{Aug. 5, 2014} \Rightarrow t=4 \quad \Delta t=2$$

$$t_0 = 0$$

$$I_0 = 26$$

$$t_1 = 2$$

$$I_1 = 26 + (10.5\sqrt{0} + 2e^{-0.1(0)})(2) = 30$$

$$t_2 = 4$$

$$I_2 = 30 + (10.5\sqrt{2} + 2e^{-0.1(2)})(2) \approx 63$$

\therefore Approximately 63 people were infected by August 5, 2014.

By exploring the initial value problem with code, students are able to investigate the behaviour of the model over a longer period (say, over months or even years) as well as increase the accuracy of their estimations (by decreasing the step size), an activity that would be unwieldy without computer technology.

```
## Euler's method

# define variables and parameters
t = []
I = []
n = 365 # number of steps
deltat = 1 # step size

# define initial values and append to arrays
t0 = 0
I0 = 26
t.append(t0)
I.append(I0)

# algorithm
for i in range(1,n+1):
    ti = t[i-1] + deltat
    Ii = I[i-1] + (10.5*t[i-1]**0.5 + 2*np.exp(-0.1*t[i-1]))*deltat # change DE in this line
    t.append(ti)
    I.append(Ii)

# output statement
print("Approximately", round(I[n]), "people were infected", t[n],"days after August 1st, 2014.")

# plot solution
plt.plot(t,I)
plt.scatter(t[n], I[n], color='r')
plt.title("Approximate Solution to an IVP using Euler's Method")
plt.xlabel("days")
plt.ylabel("# of people infected")
plt.grid()
plt.show()
```

Approximately 48758.0 people were infected 365 days after August 1st, 2014.

Moreover, using code, this algorithm can easily be extended to systems of first-order differential equations where students can explore more complex models, such as the SEIR-model, used to study the spread of the EBOLA virus during the recent epidemic or the classic predator-prey model, which investigates the dynamics between two species interacting in a common habitat.

```

## applying Euler's method to the predator prey model

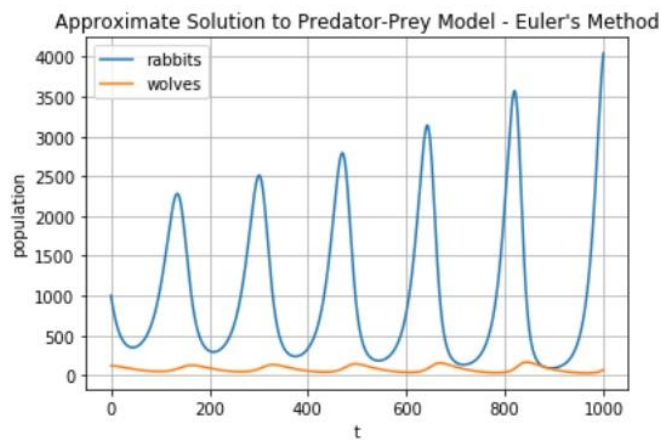
# define variables and parameters
t = []
R = [] # rabbits
W = [] # wolves
n = 1000 # number of steps
deltat = 1 # step size

# define initial values and append to arrays
t0 = 0
R0 = 1000
W0 = 120
t.append(t0)
R.append(R0)
W.append(W0)

# algorithm
for i in range(1,n+1):
    ti = t[i-1] + deltat
    Ri = R[i-1] + (0.08*R[i-1]-0.001*R[i-1]*W[i-1])*deltat # rabbit DE
    Wi = W[i-1] + (-0.02*W[i-1]+0.00002*R[i-1]*W[i-1])*deltat # wolf DE
    t.append(ti)
    R.append(Ri)
    W.append(Wi)

# plot solution
plt.plot(t,R, label = 'rabbits')
plt.plot(t,W, label = 'wolves')
plt.title("Approximate Solution to Predator-Prey Model - Euler's Method")
plt.xlabel("t")
plt.ylabel("population")
plt.legend()
plt.grid()
plt.show()

```



The project:

Together with my colleague Miroslav Lovric, I created a series of coding activities to complement the mathematical material studied in Math 1LS3, which has an enrollment of approximately 1000 students in the fall of 2018 semester. These activities are organized into four labs, which correspond to the main themes in the course: mathematical models, limits and derivatives, differential equations and integrals, and discrete-time dynamical systems. Students download these labs from the course webpage (<https://ms.mcmaster.ca/lovric/1LS3.html>) and work on them in Jupyter Notebook, a free online platform, which supports Python code. Being mindful that the course is primarily a mathematics course, we present the coding as a numerical approach to the mathematical ideas and focus on directly applying the code, rather than teaching nuances of the Python language. Brief explanations are provided as needed in the form of comments throughout the code (for example, `x = np.linspace(1, 4, 10)` # creates an array of 10 equally spaced x-values between 1 and 4). Students are encouraged to explore models, algorithms, etc. by copying, pasting, and modifying the code in each cell.

Sample activity (modified) from Lab 1 in Math 1LS3:

Exploring a Blood Alcohol Concentration Model

Most BAC calculations are based on the Widmark formula

$$C = \frac{A}{rW} - \beta t$$

where C is the blood alcohol concentration; A is the adjusted mass of alcohol consumed; r is the Widmark factor; W is the body weight; t is elapsed time since the start of drinking; and β is the elimination rate.

Exercise: By modifying the code given below (copy/paste into a new cell first!), explore how BAC depends on each of its variables.

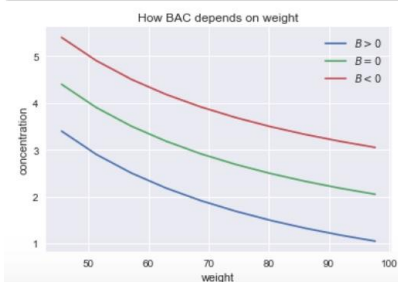
```
## BAC as a function of weight

# define parameters
A = 200 # adjusted mass of alcohol consumed
r = 1 # Widmark factor
t = 1 # time elapsed since the start of drinking
B1 = 1 # elimination rate case 1
B2 = 0 # elimination rate case 2
B3 = -1 # elimination rate case 3

# choose independent variable and define a reasonable domain
W = np.linspace(45.5,97.7,10) # body weight

# define functions
C1 = A/(r*W)-B1*t
C2 = A/(r*W)-B2*t
C3 = A/(r*W)-B3*t

# plot
plt.plot(W,C1, label = "$B>0$")
plt.plot(W,C2, label = "$B=0$")
plt.plot(W,C3, label = "$B<0$")
plt.xlabel('weight')
plt.ylabel('concentration')
plt.title('How BAC depends on weight')
plt.legend()
plt.show()
```




At the end of each lab, students are asked to respond to a series of narrative questions, which invite students to reflect on their learning experiences throughout the lab.

Sample survey questions from Lab 2:

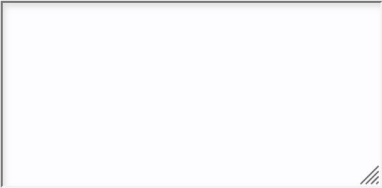
Problem #1: If you have not done so already, please read the Letter of Information (posted on our course web page) regarding the research project, Computational Thinking in Tertiary Mathematics Education, and choose one of the options below.

(A) YES, I give my consent for my responses to be used for research purposes.
(B) NO, I do not give my consent for my responses to be used for research purposes.

Problem #1: 

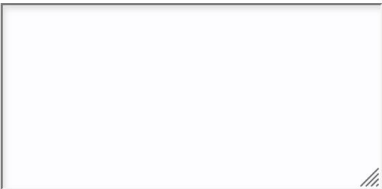
Problem #2: Describe how your perspective on a mathematical idea changed as a result of engaging with the coding activities in Lab 2.

(chars left: 500)

Problem #2: 

Problem #3: Identify an advantage that the integration of coding and mathematics can offer which traditional mathematics instruction alone cannot.

(chars left: 500)

Problem #3: 

Quotations taken from narrative responses on Lab 1:

"I think it [coding] is a great addition to Math 1LS3 because scientists often have to work with lots of data and it helps to understand how the data analysis of a software works rather than simply using it."

"One aspect that was effective was how intuitive some of the questions were because they gave the possibility to experiment with the given functions and their graphs to discover their behavior without the hassle of having to redraw the graphs by hand."

"It seamlessly integrated math and computer science into one task. Not only did it aid in deepening my math knowledge, but also enabled me to understand math through coding. The module was also not too long and very clear, allowing me to really enjoy the task at hand."

I look forward to analyzing the data at the end of the semester and contributing my research to the ongoing project, *Computational Thinking in Mathematics Education*.