Mathematics Integrated with Computers and Applications III*

An Undergraduate Programming-Based Mathematics Course for Prospective Teachers, Brock University (Canada)

Math 3P41 Project Assignments and Lab Guidelines

Dr. Chantal Buteau (Brock University)

This is part of a newly designed course that is built on the former full-year *Mathematics Integrated with Computers and Applications II-III* (MATH 2F40) course.

Since 2001, mathematics majors and prospective mathematics teachers learn through a sequence of three courses (MICA I-II-III) to design, program and use interactive environments for the investigation of math concepts, conjectures, theorems and applications. The MICA III course was recently divided into two sections; one for mathematics and science majors (MICA III) and one for prospective mathematics teachers (MICA III*), and both new courses were offered for the first time in Winter 2017.

The following material is the result of a further adaptation which occurred in the Winter 2022 semester.

Programming Languages:
- Python through *Jupyter Lab* or another IDE (same as in MICA II)
- *Visual Basic.NET* through *Visual Studio* (same as in MICA I)
- Scratch

For some information about the MICA courses, see:
- Ralph, W. (2001). Mathematics takes an exciting new direction with MICA program. Brock Teaching, 1(1), 1.
- Buteau, Muller, & Ralph (2015). Integration of Programming in the Undergraduate Mathematics Program at Brock University. In Online Proceedings of the Math + Coding Symposium, London (Canada), June 2015.

Contact:
Chantal Buteau, Brock University: cbuteau@brocku.ca
Eric Muller, Brock University: emuller@brocku.ca
Laura Broley, Brock University: lbroley@brocku.ca
MICA III* Learning Objectives:

1) To further their learning of basic methods of mathematical modelling and simulations and of experimental mathematics by use of programming as a tool (same as in MICA II and MICA III).
2) To develop an understanding of computational thinking in mathematics (learning) (unique to MICA III*).
3) To understand the place of [computational thinking/programming] in the Ontario mathematics curriculum and to [become aware of/explore/begin developing] strategies for teaching programming as a tool for mathematics learning.

Timeline of the Course:

| Week	Lecture (Assignments are written in the week they are introduced.)	Lab (Details of content of specific labs included.)
W1	**Lecture 1**	**Lab 1**: Introduction to Programming in Scratch
W2	**Lecture 2** Assignment 1: The Calendar (Intro to Programming Scratch)	**Lab 2:** The Calendar Problem in Scratch
W3	**Lecture 3**	**Lab 3:** Basics of Python Programming
W4	**Lecture 4** Assignment 2 intro: Chords in a Circle	**Lab 4:** Work on Assignment 2
W5	**Lecture 5**	**Lab 5:** Work on Assignment 2
W6	**Lecture 6** Work on Final Project	**Lab 6:** NCDSB Meeting #1 with the teachers (2 hours)
READING WEEK		
W7	**Lecture 7** Assignment 3 intro: Prey-Predator Model	**Lab 7:** Work on Assignment 3
W8	**Lecture 8**	**Lab 8:** NCDSB Meeting #2 with the teachers (1 hour) + Work on Assignment 3
W9	**Lecture 9** Assignment 4 intro: The Secret of Life… DNA Sequences!	**Lab 9:** One Task, Multiple languages, Multiple Opportunities for Learning
W10	**Lecture 10**	**Lab 10:** Work on Assignment 4
W11	*** *Final Project*: Implementation in the classroom (with NCDSB) ***	
	Lecture 11	**Lab 11:** Work on Assignment 4
W12	**Lecture 12**	**Lab 12:** NCDSB Meeting #3 with the teachers (2 hours)

MICA III* Lab 1: Introduction to programming in Scratch
MATH 3P41 Visual & Interactive Mathematics, Winter 2022
Dr. C. Buteau (Brock University)

Go to https://scratch.mit.edu. Sign in (recommended) and click on 'Create'. Add 'pen' in your menu to the left by first clicking on:

And by selecting 'Pen'. Complete the following tasks:
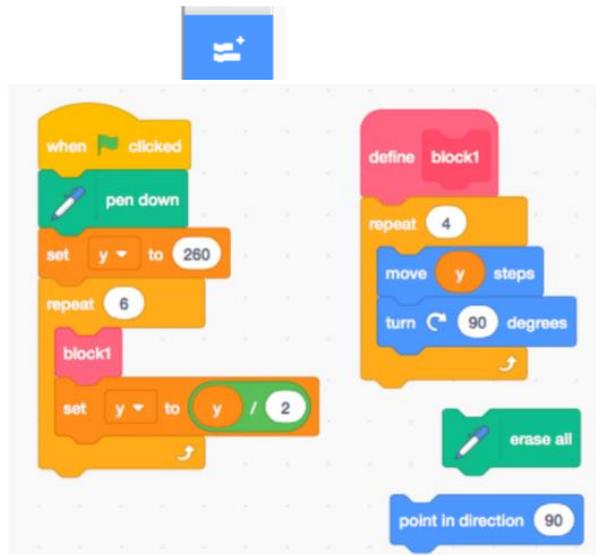
a) Look at the following code and conjecture what it will do when compiled. Then copy this code in Scratch and run it:
b) Draw an equilateral triangle
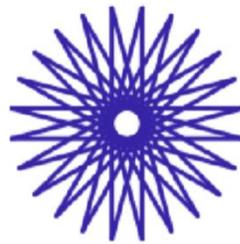c) Draw a 'flower of circles'
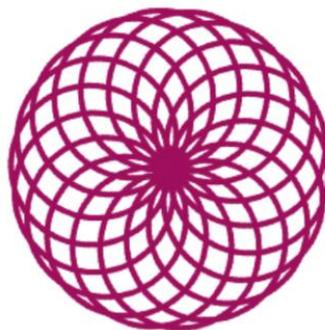
d) Read the following code and conjecture what it will do when compiled. Then copy the code in Scratch, and run it:

e) The following code draws a star as shown:

Try different values for the numbers in the code, to see the effect. Then solve the two following puzzles: Can you change only one of the numbers in the above code, to get these?
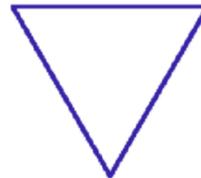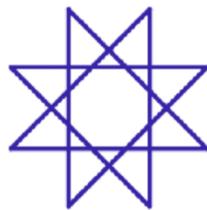
f) Optional: explore the menu in Scratch and find a few 'cool' commands.

MICA III* Lab 2: The Calendar Problem in Scratch
MATH 3P41 Visual & Interactive Mathematics, Winter 2022
Dr. C. Buteau (Brock University)

Problem:[1]

January 1, 1600, fell on a Saturday. Write a code in Scratch for the following: input is the date (day, month, year) after January 1, 1600, and output is the day of the week.

Reading input in Scratch:

Use a 'ask…and wait' Sensing command to have Sprite consecutively ask the user for the year, month, and day, e.g.:

---

[1] This question is taken from Assignment 1: The Calendar (Intro to Programming Scratch)

MICA III* Assignment #1: The Calendar (Intro to Programming Scratch)

Task guidelines created by Dr. Chantal Buteau (Brock University)

To be solved individually in Scratch:

> January 1, 1600, fell on a Saturday. Write a code in Scratch where the input is the date (day, month, year) after January 1, 1600, and the output is the day of the week.[2]

Then reflect on your experience completing the project. Briefly comment on:

    i)   Each of the 10 affordances (Gadanidis et al., 2016)[3] of programming for math learning that you experienced (or not) in your calendar project.

    ii)   The computational concepts, practices, and perspectives (Brennan & Resnick, 2012)[4] you needed (or not) to complete your project.

    iii)   A key strategy used by the instructor, TA, or a peer helper to support your completion of the project and the ways in which it supported your work or learning.

    The main part of your assignment (85%) is to be submitted by Wed. Jan. 26 by 2:30 p.m. It consists of a document with the link to your working program in Scratch, and a list of dates with weekdays that you used/tested to ensure that your calendar works properly.

    The second part of your assignment (9%) is to be submitted Fri. Jan. 28 at 4:00 p.m. It consists of your written reflection. Using size 12, Times New Roman font (2.2 cm left/right margins, single-spaced), your written document should contain 3 sections and be 1 to 1.5 pages long in total.

    The third part of your assignment (5.25%) consists of evaluating some of your peers' reflections. The rubric for this part will be made available to you by Jan. 28, the latest. This part is due on Wed. Feb. 2 at 2:30 p.m. Your feedback on the received evaluations is due Fri. Feb. 4 at 4:00 pm (0.75%).

    Note: that you may want to wait to finalize the second part of the assignment until after the lecture on Mon. Jan. 24 since, for this first assignment, we will briefly reflect together about your experience in the context of Computational Thinking.

---

[2] Problem proposed at CMESG 2016 by Miroslav Lovric (McMaster University, Canada)

[3] Gadanidis, G., Hughes, J. M., Minniti, L., & White, B. J. (2016). Computational thinking, grade 1 students and the binomial theorem. Digital Experiences in Mathematics Education, 1-20.

[4] Brennan, K., & Resnick, M. (2012, April). New frameworks for studying and assessing the development of computational thinking. In Proceedings of the 2012 annual meeting of the American Educational Research Association, Vancouver, Canada (pp. 125).

Evaluation of Assignment #1: The Calendar (Intro to Programming Scratch)

Student's ID#_____ Grade =   /100

1) The program runs and gives the correct output.

   /50

2) The selection of test dates is adequate and complete.

   /10

3) The program's visual interface is easy to understand and use and is very attractive.

   /10

4) The program's structure is well designed and makes use of blocks and comments.

   /15

5) Your reflection summary (part B–9%) is well driven, adequately addresses each topic; and your evaluation of selected peers' reflections followed the rubric and was critical and constructive (part C–5.25%); you provided adequate feedback on the received evaluations (part D–0.75%).

   /15

Any comments by the marker:

MICA III* Lab 3: Basics of Python Programming
MATH 3P41 Visual & Interactive Mathematics, Winter 2022
Dr. C. Buteau (Brock University)

Part I (70 minutes)
In your group of four, complete the 'Basics–Coding Techniques for Exploring Mathematical Concepts' worksheet activity:

- Two students 'S' (with no or little back group in Python) will work together in completing the worksheet. One student shares the screen, the other student works as well on their own computer. (Important: Do not only watch.) At any time, you can ask for help to students 'T'.
- Two students 'T' (with some knowledge of Python) will follow the progress of the two students 'S' (should also be completing it on their own computer for review purposes–keep up with students 'S'). If you deem it relevant, you can intervene to help students 'S', in addition to intervene upon their request.

To start the Python worksheet:
In https://jupyter.org/try, select JupyterLab. Upload the document 'A.Basics-3P41.ipynb'[5] (click on arrow showing up).The name of your document will then show in the list on the left: double-click on it to open. *HINT: make sure the kernel is active. Sometimes it stops and you need to reactivate it or close the file, reopen it, and run the cells.*

Part II (20 minutes)
Collective discussion and instruction about what to submit.

Part III (15 minutes)
Go back to your group of four. Complete your worksheet activity and/or prepare your report for submission.

Aim of this lab: To learn (or review) basics of programming in Python for exploring math concepts and to start developing an awareness of a task design (Use, Modify, Create (Waite & Grover, 2020)[6]) for math and coding activities.

---

[5] Please contact Dr. C. Buteau (Brock University) by email (cbuteau@brocku.ca) if you would like to inquire about this file.

[6] Waite, J., & Grover, S. (2020). Worked examples & other scaffolding strategies. In S. Grover (Eds.), Computer Science in K-12: An A-to-Z handbook on teaching programming (240-249). California, CA: Edfinity.

LAB REPORT:
Submit a Jupyter Notebook python file in which you:
1) Solve the coin simulation problem (at the bottom).
2) Create an exercise of similar format as the coin simulation. Math topic is of your choice (for Gr 9 +).

DO NOT write your names in the file.
The group leader submits the file for the whole group.
This is due Fri. Feb. 4 at 4 p.m. (one submission per group).
Then by Wed. Feb. 9 at 2:30 p.m., each individual evaluates one group's activity by following the rubric.

You will be evaluated on:
1) Working code of the coin problem simulation.
2) Scaffolding of math programming in Python from 'use', to 'modify', to 'create' (Waite & Grover, 2020)[7] stages.
3) The final stage ('create') provides an interesting perspective on the math for the learner (makes possible a simulation; finds/plots solutions; makes possible to explore some situation(s); etc.).

---

[7] Waite, J., & Grover, S. (2020). Worked examples & other scaffolding strategies. In S. Grover (Eds.), Computer Science in K-12: An A-to-Z handbook on teaching programming (240-249). California, CA: Edfinity.

MICA III* Assignment #2: Chords in a Circle
Task guidelines created by Dr. Chantal Buteau (Brock University)

Consider the following problem:

Consider a circle and an equilateral triangle inscribed in it. Pick a chord at random in the circle. What is the probability that the chord is longer than a side of the equilateral triangle?

Part A:
Create in Python a simulation of this problem that allows you to estimate the wished theoretical probability. The program should visually show the statistical experiments (in a way that could help you develop a theoretical explanation), as well as a numerical value of the calculated empirical probability. You can use Jupyter Lab or another IDE for your assignment. You may wish to use (or have a look at) the 'Assign 2 Starter.ipynb'. Then, write a short report with the following sections:

i)   Formulation of the problem.
ii)   Description of the mathematics of the problem, including any mathematics needed in the program.
iii)   A short discussion of your results.

Part B:
After you are done with Part A, reflect on your learning experience:

i)   Identify (and briefly justify) all the computational practices (according to Weintrop et al., 2016)[8] that you engaged in as you completed your task.
ii)   Comment on the ways in which programming was useful to your solving of this mathematical problem (indicate in parentheses one or more affordances that contributed to it). Do you think you could have solved it without the use of programming? Explain.
iii)   Identify a 'key' strategy (different than in your Assignment 1 reflection) used by the instructor, TA, or a peer helper to support your completion of the project and the ways in which it supported your work or learning.

Parts C-D:
The third part C of your assignment consists of evaluating some of your peers' reflections. The rubric for part C will be made available to you by Feb. 14. And finally, your feedback on the received evaluations of your reflection forms Part D of the assignment.
What to submit:

---

[8] Weintrop et al. (2016). Defining Computational Thinking for Mathematics and Science Classrooms. Journal of Science Education and Technology, 25 (1), 127-147.

- Part A of your assignment (85%) is to be submitted Wed. Feb. 16 at 3:00 p.m. in the lab. It will consist of an 8x11 envelop with your name on it that includes:
    - i) Your working program in Python on file saved on a USB.
    - ii) Your printed written report with your name, student number, and 'Assignment 2' as the heading of your written document.
- Part B of your assignment (9%) is to be submitted Fri. Feb. 18 at 4:00 p.m. It consists of your written reflection. Using size 12, Times New Roman font (2.2 cm left/right margins, single-spaced), your written document should contain 3 sections and be 1 to 1.5 page long in total. For question iii) start by mentioning which key strategy you reported on in Assignment 1 (The Calendar (Intro to Programming Scratch)). Do NOT write your name on your reflection.
- Part C of your assignment (5.25%) consisting of selected peers' reflections is due on Wed. Mar. 2 at 2:30 p.m.
- Part D or your assignment (0.75%) consisting of your feedback on the received evaluations is due Fri. Mar. 4 at 4 p.m.

Evaluation of Assignment #2: Chords in a Circle

Student's ID#_____   Grade =   /100

1) The program properly simulates random chords in the circle and shows the empirical probability. It allows to properly explore the problem.

/50

2) The program is friendly to use (can simulate many times without having to compile the program again; it is clear the kind of input to enter).

/5

3) The program's structure is well designed and makes good use of functions and comments.

/10

4) The written description of the mathematical model is clear, complete, and correct, and the discussion of the results is relevant.

/20

5) Your reflection summary is well driven and adequately addresses each topic (Part B - 9%), your evaluation of peers' reflections followed the rubric and was critical and constructive (Part C - 5.25%), and you provided adequate feedback on the received evaluations (Part D - 0.75%).

/15

Any comments by the marker:

MICA III* Assignment #3: Prey-Predator Model

Task guidelines originally created by Dr. Bill Ralph (Brock University) and adapted by Dr. Chantal Buteau (Brock University)

This assignment concerns an exploration, by use of graphical analysis, of the predator-prey model described by the Lotka-Volterra equations, namely:

$$\frac{dR}{dt} = kR - aRW$$

$$\frac{dW}{dt} = -rW + bRW$$

where *a, b, k,* and *r* are positive constants, *t* is time in months, and *R* (for 'rabbits') represents the prey population and *W* (for 'wolves') represents the predator population. As well as its modified version:

$$\frac{dR}{dt} = kR\left(1 - \frac{1}{c}R\right) - aRW$$

$$\frac{dW}{dt} = -rW + bRW$$

where c is a positive constant.

Part A – Program and Exploration

Your Program: Modify Euler's method to solve numerically either of the two systems of equations by use of a 3-dimensional dynamical system involving t_n, R_n, and W_n, and display the numerical solution as graphs (in a same plot though with two different scaling for the vertical axis) of the pair R(t) and W(t), as well as the phase trajectory for initial conditions given by the user (as input). The following gives more details about your program in Python in view of its use for investigation:

- Your program will contain two distinct parts: one for numerical solutions and one for the phase trajectory.
- Your program will initialize, at the top of your program (for easy manipulations), all the parameters involved in the model and in the graphical representations of the solutions. For example, a parameter MaxR for the maximum number of preys for the vertical axis of your prey population graph.
- The graphs of the two population functions R(t) and W(t) will be shown in a same plot using two different colors; the horizontal axis (for time *t*) will use the same scale for both graphs, but the vertical axis will use different scales (from 0 to MaxR, and from 0 to MaxW) for the two population functions. Add a legend for your colors on the display.
- The phase trajectory will be plotted for initial conditions (R_0, W_0) given by the user
- Your program will first ask the user (you!) whether to compute the original model or the modified model.

You may wish to use (or have a look at) the 'Assign3Starter.ipynb'.

Your Exploration: Using your program, answer the following questions (and provide screen shots to support your arguments) about the model of aphids (prey) and ladybugs (predator) whereby we set

k=2, a=0.01, r=0.5, and b=0.0001. We also set initial conditions to R_0=1000 and $W_0 = 200$. For graphing purposes, you may want to set MaxR=15000, MaxW=400, and h=0.001.

1. Plot the population of both species and describe how they evolve over time.
2. Determine the equilibrium points, describe what it means in the context of the modeled situation, and categorize them based on an exploration of the trajectories close to the equilibrium points: stable, unstable, or neither (i.e., periodic motions around the point).
3. Describe how the populations change when k is increased by a small amount. Explain why you would expect the model to behave this way.
4. Describe how the populations change when a is increased by a small amount. Explain why you would expect the model to behave this way.
5. What is the interval, in months, between peak aphid populations? Show how you obtained your answer.
6. If we were now to consider the modified model of these populations with c=10,000:
    a. Describe, in the absence of ladybugs, what the model would predict about aphids?
    b. How would you describe what c represents in the model?
    c. Describe, according to this model, how the populations evolve over time and how it differs from the initial model in questions 1-2.

Part B – Reflection (about 4-8 lines per feature)

After you are done with Part A, reflect on your learning experience by:

i. Identifying (and briefly explaining) all the different computational practices (according to Weintrop et al., 2016)[9] that you engaged in as you completed your project, in two parts:
    a. The practices that you also engaged in during your completion of Assignment 2.
    b. The practices that are new in this assignment.

ii. In your two first assignments, you have been reflecting on helpful instructional strategies (by the instructor, the TA, or a peer) that you experienced as a learner. In this assignment, reflect more generally on the culture of the learning environment in the lab; how would you describe the 'ideal culture', i.e., the one that would lead to the most learning for you when working on this kind of mathematical investigation project (Part A of assignments)? Identify the three most important key features and explain how they (would) help you.

What to submit:

- Part A of your assignment (85%) is to be submitted Thurs. Mar. 17 at 3:00 p.m. It consists of an 8x11 envelop with your name on it that includes:
    i) Your working program in Python on file saved on a USB.
    ii) Your printed written report (answers to questions 1-6 of your exploration) with your name, student number, and 'Assignment 3' as heading of your written document.
- Part B is to be submitted Sat. Mar. 19 at 4:00 p.m. It consists of your written reflection. Your written document should contain two sections and be 1 to 1.5 pages long. Do NOT write your name on your reflection.
- Part C, consisting of selected peers' reflections is due on Wed. Mar. 23 at 2:30 p.m.
- Part D, consisting of your feedback on the received evaluations is due Fri. Mar. 25 at 4 p.m.

---

[9] Weintrop et al. (2016). Defining Computational Thinking for Mathematics and Science Classrooms. Journal of Science Education and Technology, 25 (1), 127-147.

Evaluation of Assignment #3: Prey-Predator Model

Student's ID#_____   Grade =   /100

1) The program's structure is well designed to make it friendly to use for the exploration (e.g., parameters can easily be manipulated; easy access to either the original or the modified model; etc.) and makes good use of functions and comments.

/20

2) The program properly shows the two population graphs in a same plot (using two different scales for the vertical axis).

/20

3) The program shows the phase portraits for initial conditions selected by the user given through an input.

/20

4) Your written report containing answers to the exploration questions are clear and correct.

/25

5) Your reflection summary is well driven and adequately addresses each topic (Part B - 9%), your evaluation of peers' reflection followed the rubric and was critical and constructive (Part C - 5.25%), and you provided adequate feedback on the received evaluations (Part D - 0.75%).

/15

Any comments by the marker:

MICA III* Graded Lab 9: One Task, Multiple languages, Multiple Opportunities for Learning

MATH 3P41 Visual & Interactive Mathematics, Winter 2022
(Participation: 1%; Exit Ticket: 1%)
Dr. C. Buteau (Brock University)

Part I: Programming a predator-prey model in Excel
Download the Excel file. This is to be done together.

Part II: Exploration of the friendliness of your Excel worksheet
Alone or with a peer: *You are a mathematician. You now have an automatized computational model, and you want to use it for your mathematical inquiry (e.g., answering questions at page 2 of your assignment 3) ... how 'friendly' is your computational model?*

Part III: Predator-Prey in Scratch
Open an existing program: https://scratch.mit.edu/projects/3285161/editor/. Look into the code (e.g., disable the code from a green flag to quickly see what the procedure does). Alone or with a peer: *You are a mathematician. You have an automatized computational model, and you want to use it for your mathematical inquiry (e.g., answering questions at page 2 of your assignment 3) ... how 'friendly' is your computational model?*

Part IV: Comparing the different programming environments
Consider the following predator-prey computational models: in VB.Net, Scratch, Excel, and Python. Compare them and reflect on:
  a) Which one(s) is/are *more user-friendly to support a mathematical inquiry* about this predator-prey model in terms of
      i) providing *best output and feedback* (e.g., answering questions like in your assignment 3)
      ii) *easiness to 'see' and 'code'* (and potentially modify) *the mathematical model* and the visual representations of the solutions?
  b) In what ways are they *similar*, and in what ways are they *different*? Do they have *different affordances[10]* for this particular mathematics task (a mathematical inquiry about this predator-prey model)?
  c) If you were to prepare a project activity to code e.g., such a model with a Grade 12 math club, *which programming language*/environment would you use *based on its features and why?*
      i) *What other criteria* might you need to consider when selecting a language for the task?

---

[10] Gadanidis, G., Hughes, J. M., Minniti, L., & White, B. J. (2016). Computational thinking, grade 1 students and the binomial theorem. Digital Experiences in Mathematics Education, 1-20.

MICA III* Assignment #4: The Secret of Life…DNA Sequences!

Task guidelines originally created by Dr. Bill Ralph (Brock University) and adapted by Dr. Chantal Buteau (Brock University)

This assignment is an introduction to the important and growing new field of bioinformatics which uses mathematics to analyze biological data. Your task is to create and use a program that allows you to explore the DNA content of a living being's genome.

Part A – Program and Investigation

Your Program: Create a program in Python that calculates diverse proportions in a DNA sequence: proportions of each nucleotide (G, C, T, and A), C+G content (called the *base composition*), the paired nucleotides CG, and the trinucleotides TAA, called a *stop codon*. From a text file, the program will read from a text file the sequence (in the format found under 'Origin' at http://www.ncbi.nlm.nih.gov/nuccore/7768734), display it and it's size in a textbox, and will 'clean it' (i.e., remove spaces and numbers) before calculating the proportions.

The program will also allow a visualization of the DNA sequence as a kind of normalized walk in the plane: Start at (0,0); and read through the DNA sequence; if 'C': $+1/n_C$ in x-direction; if 'G': $-1/n_G$ in x-direction; if 'T': $+1/n_T$ in y-direction; and if 'A': $-1/n_A$ in y-direction, e.g. where n_A is the number of A in the DNA sequence. The user may also decide to display in the same plane (using a different color) a normalized, random walk of same length, for comparison purposes.

More about the DNA sequence files: Go to http://www.ncbi.nlm.nih.gov/nuccore/7768734. Either use this data in the search box at the top, or type in the name of any living thing in which you are interested, then scroll down and copy the small amount of DNA that is under the word "Origin". Paste this sample including the numbers and spaces into notepad and save it. This file should appear on your computer as FileName.txt. If you use Spyder IDE in Anaconda, you can use the following code to read the file content and save it into a string:

```
from tkinter import Tk
from tkinter.filedialog import askopenfilename #pop-up should appear asking the user to choose a file (note I had to minus the python screen to see it)
Tk().withdraw()
filename = askopenfilename()
with open(filename,'r') as f:
    contents=f.read()
```

If you want to use Jupyter Lab, this import will not work. What you can do instead, is modify the content of FileName.txt by removing all line breaks in Microsoft Word: save the whole content, select Find and Replace, and type in Find: ^p, type in Replace: ,. This will replace all the line breaks with commas. Copy the content, and paste it in a cell in Jupyter Lab, including a comment that labels the string, as:

```
# genome from a Human Being
contents="paste here"
```

Your Investigation: Use your program to answer the following questions in a written report.

1. Use the **Human DNA sequence** from: https://www.ncbi.nlm.nih.gov/nuccore/AC073210.8. We will first question the independence of the nucleotides G and C present in the DNA sequence by exploring the 'CG' content. In the following, use the relative frequency of each nucleotide as probability that a nucleotide selected at random is of that type.
   a. Compare the proportion of 'CG' content with the (hand-calculated) conditional probability that a nucleotide chosen at random is G given that the previous one in the sequence is a C. What does it suggest: *that the nucleotide G and C in the DNA sequence are occurring randomly and independently? Explain*.
   b. Conduct a statistical test on proportions to question whether the nucleotides C and G are independent by determining whether the observed proportion of 'CG' in the DNA sequence is significantly different from the expected proportion if C and G were occurring randomly and independently. *What do you conclude?*
   c. Have also a look at the stop codon TAA. *Does it occur more or less often than you would expect by chance* if you assume that the nucleotides are distributed randomly and independently? *Explain. HINT: Suppose for example, that your DNA string was catggacat. Then the two letter strings would be ca, at, tg, gg, ga, ac, ca and at so the probability of picking the two-letter string ca at random would be 2/8.*
2. Look at the DNA sequence of five other living beings.
   a. Compare the proportions of their nucleotides, and of their base composition (G+C content). Summarize the results (remember to indicate the name of the living being) and *comment on whether these proportions are more or less constant for the different types of life you have examined*.
   b. Research showed that the frequency of occurrence of TAA stop codon is correlated to the GC-content. What does your data suggest: a positive or negative correlation? Justify you answer. Write a sentence (to a fictive Grade 10 student) that describes what this means. BONUS: add a visualization of the correlation (could be in your Python code or using a different technology).
   c. For each living being, generate their "DNA sequence normalized walk" visualization. Compare them; *do they seem different to you*? Provide screenshots.
   d. *Do the DNA sequence normalized walks in c) look random to you?* Compare them with normalized random walks of the same length. Provide screenshots and comment.

PART B – The Reflection
Write a reflection on your experience completing the project:
i. Consider the cognitive, interpersonal, and intrapersonal domains described by Vivian, Grover, & Falkner (2020)[11]. Illustrate how each domain was part of your experience in completing this project (provide one example for each domain).
ii. Describe one key moment where you received or gave help: What was the issue, what was the strategy used by the 'helper', and in what ways did it help the person who needed help (in terms of cognitive, interpersonal, and/or intrapersonal domains)?

What to submit:

---

[11] Vivian, R., Grover, S., & Falkner, K. (2020). Knowledge, skills, attitudes, and beliefs: Learning goals for introductory programming. In S. Grover (Eds.), Computer Science in K-12: An A-to-Z handbook on teaching programming (113-124). California, CA: Edfinity.

- Part A of your assignment (85%) is to be submitted Wed. Apr. 6 at 3:15 p.m. This consists of an 8x11envelop with your name on it that includes:
    o Your working program in Python on file saved on a USB, together with the genome files for different species, if relevant.
    o Your printed written report (answers to questions of Your Investigation) with your name, student number, and 'Assignment 4'as heading of your written document.
- Part B of your assignment is to be submitted Fri. Apr. 8 at 4:00 p.m. It consists of your written reflection. Use size 12, Times New Roman font (2.2 cm left/right margins, single-spaced). Your written document should contain two sections and be 1-1.5 pages long in total. Do NOT write your name on your reflection.
- Part C of your assignment consisting of selected peers' reflections is due on Sun. Apr. 10 at 4:00 p.m.
- Part D of your assignment, consisting of your feedback on the received evaluations is due Tues. Apr. 12 at 4 p.m.

Evaluation of Assignment #4: The Secret of Life... DNA Sequences!

1) Part A–The program

/40

- The program reads the content of a file and calculates all proportions needed for the numerical investigation.   /20
- It displays the 'DNA' walk.   /6
- It displays a random walk of the same length.   /6
- The program is well structured and has useful comments, where relevant, and the output supports the investigation well.   /8

2) Part A–The Investigation

/45

- Question 1 Part a) /4; Part b) /5; Part c) /4
- Question 2 Part a) /8; Part b) /8; Part c) /8; Part d) /8

3) Part B-D–The Reflection

/15

- The reflection summary is well driven and adequately addresses each topic (Part B - 9%), the evaluation of peers' reflection followed the rubric and was critical and constructive (Part C - 5.25%), and the student provided adequate feedback on the received evaluations (Part D - 0.75%).

Any comments by the marker:

Final Project Overview
MICA III* Visual and Interactive Mathematics, Winter 2022
Dr. C. Buteau (Brock University)

Your final project concerns selecting (or developing) and implementing in a school classroom a programming-based mathematics activity done in collaboration with a schoolteacher (or two) from NCDSB. It will involve also reflecting on your experience, as well as an oral presentation.

Timeline summary:

-Mon. Feb. 14	List of teachers/schools, with Grade levels available + Etiquette when working with your associate teacher
-Wed. Feb. 16	Meeting with NCDSB teachers
-Fri. Mar. 4	Draft 1 (1%) - Activity description: submission due by 4 p.m.
-Wed. Mar. 9	Meeting with your associate teacher (1hr) during lab
-Mon. Mar. 21	Draft 2 (2%)–Enhanced description: submission due by 4 p.m.
-Between Mar. 28 and Apr. 1	*Implementation of your activity in the classroom*
-Wed. Apr. 6	Reflection with NCDSB teachers
-Mon. Apr. 11	Final project report (22%) submission due at 4 p.m.
-Thurs. Apr. 14	Oral presentation (5%) @8:30 a.m.- 12:00 p.m.

Three possible approaches (types):
   A. Design of an activity from scratch.
   B. Use and modification of an activity from a resource.
   C. Use of an activity from a resource.
Also: some student teams are matched with two teachers (possibly 2x the implementation). Some teams might go to the classroom more than once for their activity.

Oral Presentation: Thurs. Apr. 14 @8:30 a.m.- 12:00 p.m. (5%)
Short presentation (1 per team) of your activity, implementation, and reflections. 6-8 min–3.5%
Your evaluation of two other teams' presentations–1.5%

Final Project Submission: Mon. Apr. 11 (22%)
Your final project report contains two parts for a project of type A: the activity summary with appendices, and the reflection summary. The marking scheme respects these two components, in addition to a third one: fulfillment of roles for the community partnership.
Your final project report contains three parts for a project of type B or C: the implementation report, your analysis of the implementation using the literature, and the reflection summary. The marking scheme respects these three components, in addition to a fourth one: fulfillment of roles for the community partnership.

More details below.

Type A–*designing an activity from scratch*

The report contains 2 parts, the activity summary with appendices and the reflection summary:

1) Your *Activity Summary with Appendices*[12] (2-3 pages + appendices) will include the following sections (and including these as headlines):

    - **Name of the activity**
    - **Grade level**
    - **Learning objectives**, with links to the curriculum expectations (for math topic(s) and for coding). Clearly outline all of them (possibly distinguishing primary and secondary objectives)
    - **Activity instructions**
    - **Classroom implementation**. Description of how the activity can be integrated in the classroom (possibly different scenarios). Include proposed time.
    - **Discussion points**. List of recommended (at least 3) questions for the class
    - **Things to note** (if applicable). Notes for the teachers.
    - **Agreement statement** to participate or not to the collective class project and have the Activity Summary posted online on the MKN website.

    And Appendices:
    - **Student guidelines**
    - **File(s)** if relevant
    - A **complete solution**, including comments/recommendations to the teachers, possibly based on your implementation experience (e.g., *when drawing an equilateral triangle, a common first approach is to make Sprite turn 60 degrees instead of the supplementary angle 120 degrees*.). Includes also comments to highlight key part(s) of the activity (and why they are 'key' parts) and comments pertaining to some Computational Thinking Affordances[13].

2) Your **Reflection Summary** about your experience will include the following sections:

    - **Associate class (community partnership).** A short description explaining/describing what led you to the proposed activity. Highlight your associate teacher(s)' needs and goals. (8-15 lines)
    - **About the activity implementation**. How did the implementation go? As planned? Any surprise. Include anecdotes (use pseudonyms). If the implementation led you to modify your activity summary, please specify what and why. (6-15 lines)
    - **Two major take-aways.** Select two ideas shared in the collective reflection on April 6 and reflect on how it connected to your experience. (6-15 lines)
    - **What you have learned personally.** Reflect back on your personal goals prior to the implementation. (each student: 8-15 lines)

Grading Scheme (Type A). Names: _____ Grade: ____

---

[12] Note that you may revise/update/improve your activity summary based on what you have learned from the implementation of your activity.

[13] Gadanidis, G., Hughes, J. M., Minniti, L., & White, B. J. (2016). Computational thinking, grade 1 students and the binomial theorem. Digital Experiences in Mathematics Education, 1-20.

Activity:

/75

1) Meaningful use of the potential of programming for mathematics learning/application (incorporates many of the Computational Thinking Affordances[14] for math learning).

/10

2) Math content/topic addressed in depth.

/10

3) The activity aligns with the identified learning objectives.

/10

4) Guidance to students (if appropriate; if not, the weight is added to the next bullet): complete, detailed description.

/10

5) Guidance to teachers: complete, clear.

/15

6) The activity is engaging for all learners in the class (low floor, high ceiling![12]).

/10

7) Communication of mathematics in the documents.

/10

Reflection:

/15

a) About the implementation (insightful; good choice of anecdote, if relevant).

/6

b) Two major take-aways (relevance, depth).

/5

c) What you have learned personally (relevance, depth), with individual answers.

/2

d) Quality of writing and clarity: no spelling, grammar, and punctuation errors. Ideas are presented in a clear and organized manner.

/2

Fulfillment of roles for the community partnership:

/10

- Met deadlines and accomplished the expected work, met with the teachers when expected, was attentive to the associate teacher(s)' needs and goals, was on time and professional, respectful of the existing classroom culture. *This grade will be based on comments provided by the associate teacher(s) and Numeracy Consultants, if relevant; on the TA's and professor observations; and on your peer's short comments that will be submitted separately in the google form, and the first part of your reflection.*

Types B & C– *use (and modification) of an activity from a resource*

---

[14] Gadanidis, G., Hughes, J. M., Minniti, L., & White, B. J. (2016). Computational thinking, grade 1 students and the binomial theorem. Digital Experiences in Mathematics Education, 1-20.

1. Your *Activity Implementation Report Summary* (3-5 pages), will include the following sections (and including these as headlines):
   - **Name of the activity**: 'Implementing ....'
   - **Reference of selected resource**
   - **Grade level**
   - **Learning objectives**, with links to the curriculum expectations (for math topic(s) and for coding). Clearly mention what is taken from the resource, and what is new based on your implementation
   - **Activity summary**: in 3-4 lines
   - **Adaption/Modification of Activity, and Rational**
   - **Report on classroom implementation (tell the story!)**: Description of how the activity was integrated in the classroom. Include, but not limited to:
     - Time(s)
     - Guiding questions to the whole class (indicate which of those were from the resource)
     - Anecdotes (use pseudonyms), highlighting key moments of (individual or collective) math and/or coding learning, of students' and/or teachers' challenges, and of some observed Computational Thinking Affordances[15] for math learning
     - Any surprises? Anything unexpected?
     - The use/usefulness of the existing teacher guidelines from the resource (i.e., hints, tips for the teacher)
   - **Our recommendations**: Notes for other teachers based on your implementation, including possibly changes to the implementation and adaptation of the activity.
   - **Agreement statement** to participate or not to the collective class project and have the Activity Summary posted online on the MKN website.
   - *Appendices* (if relevant for other teachers)

2. Your *Analysis of implementation report using the literature*. (~1-2 pages)

3. Your *Reflection Summary* about your experience will include the following sections:
   - **Associate class (community partnership)**. A short description explaining/describing what led you to select the proposed activity and your implementation. Highlight your associate teacher(s)' needs and goals. (8-15 lines)
   - **Two major take-aways**. Select two ideas shared in the collective reflection on April 6 and reflect on how it connected to your experience. (6-15 lines)
   - **What you have learned personally**. Reflect back on your personal goals prior to the implementation. (each student: 8-15 lines)

Grading Scheme (Type B or C). Names: _____ Grade: ____

Activity Implementation Report:

/70

---

[15] Gadanidis, G., Hughes, J. M., Minniti, L., & White, B. J. (2016). Computational thinking, grade 1 students and the binomial theorem. Digital Experiences in Mathematics Education, 1-20.

1. Meaningful adaptation of the resources.   /10
2. Implementation: /40
    - Description is well organized, complete, and clear.   /10
    - Key moments provide interesting, useful, and deep insights into the activity.
      /25
    - Indicated use/usefulness of teacher guidelines are relevant.   /5
3. Recommendations: /20
    - Takes a critical perspective on the implemented activity (if no or only minor change argues why not).   /12
    - Relevant and practical to other teachers.   /8

Analysis:

/15

1. Makes use of at least three relevant literature references, with complete citations at the end of this section.

/2

2. Demonstrates a good understanding of the referenced literature and makes insightful connections between reported key experiences and theory.

/11

3. Quality of writing and clarity: no spelling, grammar, and punctuation errors. Ideas are presented in a clear and organized manner.

/2

Reflection:

/5

Discusses the observations and provide a critical analysis of the task/activity in the context of learning mathematics through a computational thinking approach
- Two major take-aways (relevance, depth).   /3
- What you have learned personally (relevance, depth), individual answers.   /2

Fulfillment of roles for the community partnership:

/10

Met deadlines and accomplished the expected work, met with the teachers when expected, was attentive to the associate teacher(s)' needs and goals, was on time and professional, respectful of the existing classroom culture. *This grade will be based on comments provided by the associate teacher(s)and Numeracy Consultants, if relevant; on the TA's and professor's observations; and on your peer's short comments that will be submitted separately in the google form.*