Title: Computational and Mathematical Biology in HPC Science Gateways

SGX3 award # 2231406

Revised Course Description

In some cases, it is too dangerous or impossible to do an experiment, so we can do numerical experiments through mathematical modeling and simulation. Besides learning mathematical modeling, the students in this course will learn basic commands, syntaxes, and fundamental programming in Python and use them for solving problems in biology. The course targeted students having major in mathematics, and biology and chemistry with minor in mathematics who are interested in learning computational and mathematical biology. The course consists of 3 parts: 1) fundamental programming in Python, 2) computational biology, and 3) mathematical biology.

Use Cases

1) By using Python, determine the relative frequencies of constituent amino acids for each protein secondary structural class from the Protein Data Bank (PDB). See the Datasets column.
2) By using Python, research the best fit model for the data from in vitro experiments in which nanoparticles induced into cancer cells in biology laboratory at Jarvis Christian University, Hawkins, Texas.

Resource Needs/List

1) **Google Colab.** The students use it for programming in Python to introduce the students with Python coding without installing Python in their computers. https://colab.research.google.com/
2) **Jupiter Notebook.** The students may use TACC and Jetstream2 Resources for using the Jupiter Notebook. https://www.tacc.utexas.edu/ https://jetstream-cloud.org/
3) **Anaconda Navigator.** The students may experience installing it in their own computers. The link is this: https://docs.anaconda.com/free/navigator/index.html
4) **Python.** The students need to learn this popular language to solve problems in biology. This is the link: https://www.python.org/
5) **SciPy.** The students need to learn algorithms in solving biological problems. This is the link: https://scipy.org/
6) **scikit-learn (sklearn).** Simple and efficient tools for machine learning. They students may use it in the future. The link is this: https://scikit-learn.org/stable/
7) **GitHub.** The students need to collaborate in solving biological problems, so they can use it to share codes. The link to our GitHub repository is this: https://github.com/wsamyono/BulldogTeamFacHackGW23

Your feedback is welcome!

Sample HPC/Gateways Exercise

Exercises 7 and 12 are from this journal paper: https://doi.org/10.1371/journal.pcbi.1004867

Exercise 7:
Consider the fermentation of glucose into ethanol: $C_6H_{12}O_6 \rightarrow 2C_2H_5OH + 2CO_2$. A fermenter is initially charged with 10,000 liters of feed solution and the rate of carbon dioxide production is measured by a sensor in moles/hour. At t = 10, 20, 30, 40, 50, 60, 70, and 80 hours, the CO_2 generation rates are 58.2, 65.2, 67.8, 65.4, 58.8, 49.6, 39.1, and 15.8 moles/hour, respectively. If each reading represents the average CO_2 production rate over the previous ten hours, calculate the total amount of CO_2 generated and the final ethanol concentration in grams per liter. Note that Supplemental Chapters 6 and 9 might be useful here.

Exercise 12 (cumulative project):
First, obtain a set of several hundred protein structures from the PDB, as plaintext **.pdb** files (the exact number of entries is immaterial). Then, from this pool of data, determine the relative frequencies of the constituent amino acids for each protein secondary structural class; use only the three descriptors "helix," "sheet," and, for any AA not within a helix or sheet, "irregular." (Hint: In considering file parsing and potential data structures, search online for the PDB's file-format specifications.) Output your statistical data to a human-readable file format (e.g., comma-separated values, **.cvs**) such that the results can be opened in a statistical or graphical software package for further processing and analysis. As a bonus exercise, use Python's **matplotlib** package to visualize the findings of your structural bioinformatics analysis.

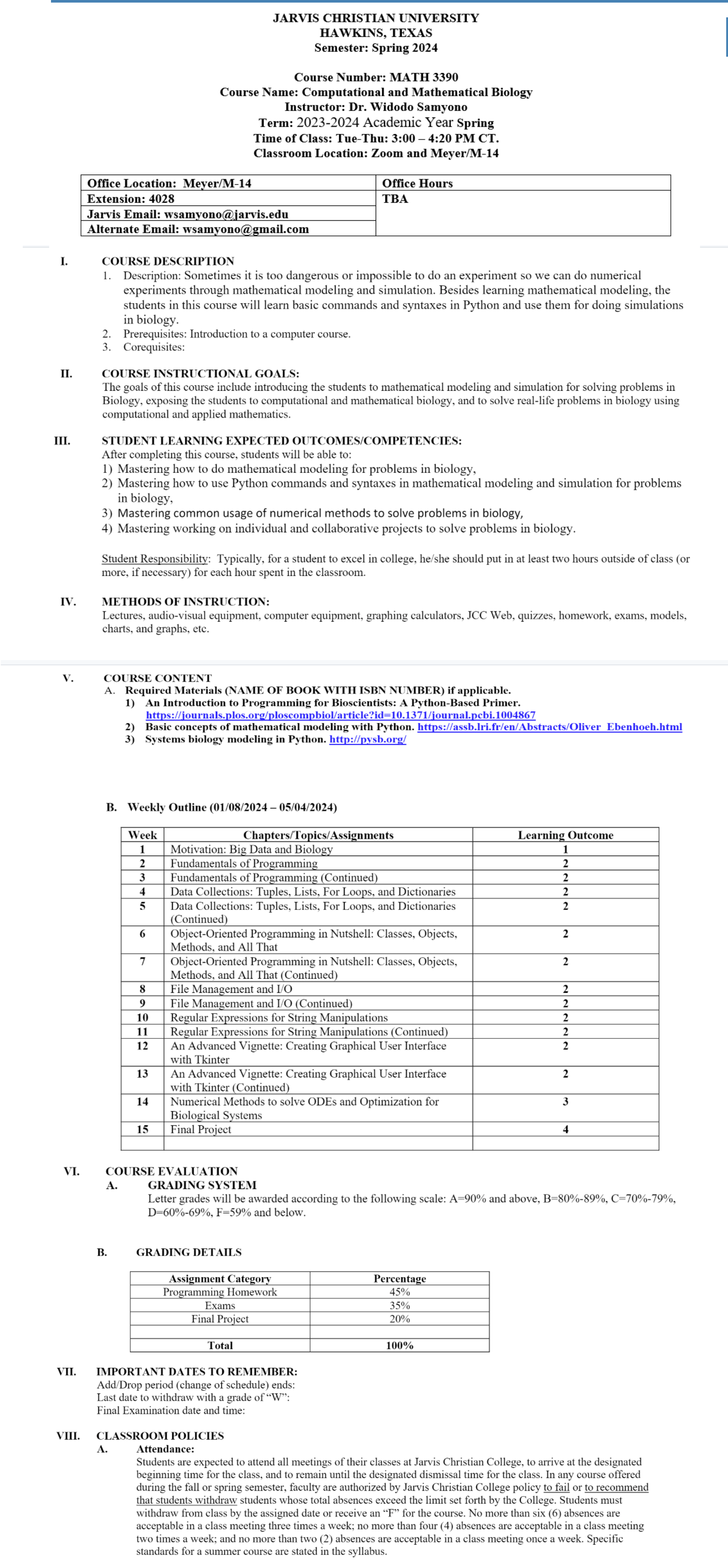
We created Exercise 13 by ourselves.

Exercise 13: Use the provided data from the in vitro experiments, by which the biology student inducing nanoparticles into cancer cells, to do mathematical modeling for predicting the cancer growths and observing the effects of nanoparticles for the cancer treatments and responses. Use the classical mathematical models from the literatures. To solve the problem, set the problems as differential equations constrained optimization problems that we learned from the class. Use the code template to start the Python coding.

Implementation Schedule

Spring 2024, January 8, 2024 – May 3, 2024.
Summer 2024, June 3, 2024 – June 28, 2024.
Spring 2025, January 6, 2025 – May 2, 2025.
Summer 2025, January 2, 2025 – June 27, 2025.

Gateway Community Mentor Syllabus Suggestions



JARVIS CHRISTIAN UNIVERSITY
HAWKINS, TEXAS
Semester: Spring 2024

Course Number: MATH 3390
Course Name: Computational and Mathematical Biology
Instructor: Dr. Widodo Samyono
Term: 2023-2024 Academic Year **Spring**
Time of Class: Tue-Thu: 3:00 – 4:20 PM CT.
Classroom Location: Zoom and Meyer/M-14

Office Location: Meyer/M-14	**Office Hours**
Extension: 4028	**TBA**
Jarvis Email: wsamyono@jarvis.edu	
Alternate Email: wsamyono@gmail.com	

I. COURSE DESCRIPTION
1. Description: Sometimes it is too dangerous or impossible to do an experiment so we can do numerical experiments through mathematical modeling and simulation. Besides learning mathematical modeling, the students in this course will learn basic commands and syntaxes in Python and use them for doing simulations in biology.
2. Prerequisites: Introduction to a computer course.
3. Corequisites:

II. COURSE INSTRUCTIONAL GOALS:
The goals of this course include introducing the students to mathematical modeling and simulation for solving problems in Biology, exposing the students to computational and mathematical biology, and to solve real-life problems in biology using computational and applied mathematics.

III. STUDENT LEARNING EXPECTED OUTCOMES/COMPETENCIES:
After completing this course, students will be able to:
1) Mastering how to do mathematical modeling for problems in biology,
2) Mastering how to use Python commands and syntaxes in mathematical modeling and simulation for problems in biology,
3) Mastering common usage of numerical methods to solve problems in biology,
4) Mastering working on individual and collaborative projects to solve problems in biology.

Student Responsibility: Typically, for a student to excel in college, he/she should put in at least two hours outside of class (or more, if necessary) for each hour spent in the classroom.

IV. METHODS OF INSTRUCTION:
Lectures, audio-visual equipment, computer equipment, graphing calculators, JCC Web, quizzes, homework, exams, models, charts, and graphs, etc.

V. COURSE CONTENT
A. **Required Materials (NAME OF BOOK WITH ISBN NUMBER) if applicable.**
1) **An Introduction to Programming for Bioscientists: A Python-Based Primer.** https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1004867
2) **Basic concepts of mathematical modeling with Python.** https://assb.lri.fr/en/Abstracts/Oliver_Ebenhoeh.html
3) **Systems biology modeling in Python.** http://pysb.org/

B. **Weekly Outline (01/08/2024 – 05/04/2024)**

Week	Chapters/Topics/Assignments	Learning Outcome
1	Motivation: Big Data and Biology	1
2	Fundamentals of Programming	2
3	Fundamentals of Programming (Continued)	2
4	Data Collections: Tuples, Lists, For Loops, and Dictionaries	2
5	Data Collections: Tuples, Lists, For Loops, and Dictionaries (Continued)	2
6	Object-Oriented Programming in Nutshell: Classes, Objects, Methods, and All That	2
7	Object-Oriented Programming in Nutshell: Classes, Objects, Methods, and All That (Continued)	2
8	File Management and I/O	2
9	File Management and I/O (Continued)	2
10	Regular Expressions for String Manipulations	2
11	Regular Expressions for String Manipulations (Continued)	2
12	An Advanced Vignette: Creating Graphical User Interface with Tkinter	2
13	An Advanced Vignette: Creating Graphical User Interface with Tkinter (Continued)	2
14	Numerical Methods to solve ODEs and Optimization for Biological Systems	3
15	Final Project	4

VI. COURSE EVALUATION
A. GRADING SYSTEM
Letter grades will be awarded according to the following scale: A=90% and above, B=80%-89%, C=70%-79%, D=60%-69%, F=59% and below.

B. GRADING DETAILS

Assignment Category	Percentage
Programming Homework	45%
Exams	35%
Final Project	20%
Total	**100%**

VII. IMPORTANT DATES TO REMEMBER:
Add/Drop period (change of schedule) ends:
Last date to withdraw with a grade of "W":
Final Examination date and time:

VIII. CLASSROOM POLICIES
A. Attendance:
Students are expected to attend all meetings of their classes at Jarvis Christian College, to arrive at the designated beginning time for the class, and to remain until the designated dismissal time for the class. In any course offered during the fall or spring semester, faculty are authorized by Jarvis Christian College policy to fail or to recommend that students withdraw students whose total absences exceed the limit set forth by the College. Students must withdraw from class by the assigned date or receive an "F" for the course. No more than six (6) absences are acceptable in a class meeting three times a week; no more than four (4) absences are acceptable in a class meeting two times a week; and no more than two (2) absences are acceptable in a class meeting once a week. Specific standards for a summer course are stated in the syllabus.

Resources / Science Gateways

1.SGX3 Science Gateways https://sciencegateways.org/education-training/resources#/ We may use some resources for training the students.
2.Design Safe Gateways https://www.designsafe-ci.org/ We may use some of the modules to expand our course.
3.Texas Advanced Computing Center (TACC) https://www.tacc.utexas.edu/ We may use the machine to run the students' codes.
4.Jetstream2 https://jetstream-cloud.org/index.html We use the resources for running Jupiter Notebook.
5.ACCESS https://access-ci.org/ We may use the platform to collaborate with other faculty from different institutions.

Datasets

1.RCSB PDB Protein Data Bank: https://www.rcsb.org/
2.Genomic Data Commons Data Portal: https://portal.gdc.cancer.gov/
3.Data from students in Biology conducting in vitro experiments by inducing nanoparticles into cancer cells. Data can be acquired directly from the students and the biology faculty members.
4.Other biology data from Science Gateways.

Possible Expansions

The course could be expanded to more advanced topics in the future by adding some more modules and new recent science gateways resources, and literatures.
We may collaborate with other faculty from different institutions to teach and conduct research in computational and mathematical biology.

Authors

Team Member
Widodo Samyono, PhD
Mathematics Department,
Jarvis Christian University
wsamyono@jarvis.edu

HPC/Gateways Mentor
jpowell@tacc.utexas.edu
Texas Advanced Computing Center