

Math 340, Spring 2026  
Programming in Mathematics  
Instructor: Dr. Bo-Wen Shen

email: [sdsu.math340.shen@gmail.com](mailto:sdsu.math340.shen@gmail.com)

URL: <https://bwshen.sdsu.edu>

(Last Updated: 2026/01/16)

## COURSE INFORMATION

Class Days: MW  
Class Times: 9:00-9:50 am  
Class Location: HH-221

Dr. Bo-Wen Shen:  
Office Hours Times (and by appointment): 2:30-3:30 pm MWF  
Office Hours Location: GMCS 569 or Zoom

## Course Overview

### Course Descriptions:

Introduction to programming in mathematics. Modeling, problem solving, visualization. This course covers essential programming foundations and numerical methods, including:

#### Programming Foundations in Python:

- Syntax, Data types, and Control Structures
- Graphical Representation and Visualization using Matplotlib
- Array and Matrix Computing with NumPy
- Symbolic Mathematics with SymPy
- Computer Algebra with NumPy, SymPy, and SciPy.
- PyTorch (automatic differentiation) for Deep Learning (e.g., Feedforward Network and Transformer)

#### Numerical Methods and Concepts:

- Error Analysis
- Sequences and Series and Their Representation Using Numerical Arrays
- Numerical Differentiation and Finite Difference Methods for Ordinary Differential Equations (ODEs)
- Difference Equations and Matrix Methods
- Interpolation and Splines
- Numerical Integration Techniques (and Root Finding)

Upon successful completion, students will possess a strong foundation in Python programming and numerical methods, enabling them to apply computational techniques to solve diverse mathematical, scientific, and engineering problems.

### Optional Concepts (Depending on Available Time):

- Various functionalities of the SciPy package for advanced numerical computing
- Object-oriented programming principles in Python
- Exposure to parallel computing using MPI4PY
- ParaView for data visualization
- Principal Component Analysis (PCA) for dimensionality reduction

## Student Learning Outcomes:

- **Outcome 1 for Programming Foundations in Python:**
  - Develop a solid understanding of Python programming, including syntax, data types, and control structures.
  - Explore graphical representation and visualization techniques using Matplotlib.
  - Acquire practical skills in numerical computing with NumPy.
  - Apply symbolic mathematics through SymPy for problem-solving (e.g., backpropagation)
  - Apply automatic differentiation (AD) of PyTorch for backpropagation in neural networks and transformers
  - Gain proficiency in computer algebra using NumPy, SymPy and SciPy.

Course Activity: Lectures & Lab Exercises

Assessment Strategy: Homework; midterm exams, and final exam

- **Outcome 2 for Numerical Methods and Concepts:**
  - Conduct error analysis and estimate errors in numerical computations.
  - Investigate sequences and series, representing them using numerical arrays.
  - Understand sequences and time series data using the Fibonacci sequence and a simple ordinary differential equation
  - Implement numerical differentiation and finite difference methods for solving differential equations.
  - Understand the similarity and differences of the discrete and continuous forms of the Logistic equation
  - Explore various types of numerical solutions, including chaotic solutions, in the Logistic equation
  - Solve difference equations and matrix problems, emphasizing applications in boundary value problems.
  - Understand interpolation and splines for approximating functions.
  - Apply numerical integration techniques, including adaptive methods and least square regression.
  - Explore root-finding methods and understand their rate of convergence.

Course Activity: Lectures & Lab Exercises

Assessment Strategy: Homework; midterm exams, and final exam

## Relation to Other Courses:

Students may further take advanced courses such as partial differential equations (PDEs, Math531), nonlinear dynamics (e.g., Math538, Math638), applied Fourier analysis (Math668), computational PDEs (e.g., Math693A, B).

## Enrollment Information

MATH 151 and MATH 245 with a grade of C (2.0) or better in each course. Proof of completion of prerequisite(s) required: Copy of transcript. Not open to students with credit in Mathematics 242. Students with special requests may take this course subject to approval by the instructor of the course.

## Course Materials (optional)

- Langtangen, H. P., 2016: *A Primer on Scientific Programming with Python*. Springer-Verlag Berlin Heidelberg. 914pp.
- Burden, R., D. Faires, and A. M. Burden, 2014: *Numerical Analysis*. Cengage Learning. ISBN 978-1-305-25366-7.
- Gerald P. O and C. F. Wheatley, *Applied Numerical Analysis*. Pearson Education, 2004, ISBN 8131717402, 609 pp.

## Course Structure and Conduct

Style of the Course: (1) Lectures/Discussions with Jupyter notebook files, PowerPoint slides and pdf materials; (2) practices on personal computers.

## Course Assessment and Grading Policy

- Homework 20% Weekly, due at 11:00 pm Friday
- In-class Assessments 2 \*10%
- Mid Term Exam 20%
- Final Exam 33% (for Parts A and B)
- Attendance and discussions 2% (for Lectures)  
5% (for lab activities)
- Class Attendance: Students are required to attend all class meetings and discussions. Class attendance will be taken randomly.
- Late submission and make-up exams: Missed timed homework or exams can only be made up in the case of a University-approved absence.

## Course Grade

You will be guaranteed the following grades as given by your percentage score on the homework, midterm, and final project presentation and report.

A 90%	B 80%	C 70%	D 60%
A- [89%, 90%)	B+ [85%, 89%)	C+ [75%, 79%)	D+ [65%, 69%)
	B [80%, 85%)	C [70%, 75%)	D [60%, 65%)
	B- [79%, 80%)	C- [69%, 70%)	D- [59%, 60%)

## Important Dates

Jan 20 (Tue)	First day of Classes
Feb 20 (Fri)	ICA-1 (In-class Assessments 1*)
Mar 4 (Wed)	Midterm Exam
Mar 27 (Fri)	ICA-2 (In-class Assessments 2*)
Mar 30- Apr 4	Spring break
May 6 (Wed)	Last day of classes before final examinations
May 11 (Mon), 8:00-10:00	Final Exam (#)
May 21 (Thur)	Grades due from instructors (11 p.m. deadline)

\*Please refer to Lecture#1 slides for details regarding the In-class Assessments (ICA). If you have specific questions or need additional information about any of these dates, please check with the instructor.

# The final exam is divided into two sections: Part A, which involves submitting a report that summarizes homework assignments with a focus, and Part B, which is a written exam. Part A must be submitted by 11:59 pm on **May 6 (Wed)**, and Part B will take place at the scheduled time on **May 11 (Mon)**.

## University Policies

**Accommodations:** If you are a student with a disability and are in need of accommodations for this class, please contact Student Ability Success Center at (619) 594-6473 as soon as possible. Please know accommodations are not retroactive, and I cannot provide accommodations based upon disability until I have received an accommodation letter from Student Ability Success Center. Your cooperation is appreciated.

**Student Privacy and Intellectual Property:** The Family Educational Rights and Privacy Act (FERPA) mandates the protection of student information, including contact information, grades, and graded assignments. I will not post grades or leave graded assignments in public places. Students will be notified at the time of an assignment if copies of student work will be retained beyond the end of the semester or used as examples for future students or the wider public. Students maintain intellectual property rights to work products they create as part of this course unless they are formally notified otherwise.

**Religious observances:** According to the University Policy File, students should notify the instructors of affected courses of planned absences for religious observances by the end of the second week of classes.

**Medical-related absences:** Students are instructed to contact their professor/instructor/coach in the event they need to miss class, etc. due to an illness, injury or emergency. All decisions about the impact of an absence, as well as any arrangements for making up work, rest with the instructors. Student Health Services (SHS) does not provide medical excuses for short-term absences due to illness or injury. When a medical-related absence persists beyond five days, SHS will work with students to provide appropriate documentation. When a student is hospitalized or has a serious, ongoing illness or injury, SHS will, at the student's request and with the student's consent, communicate with the student's instructors via the Vice President for Student Affairs and may communicate with the student's Assistant Dean and/or the Student Ability Success Center.

**SDSU Economic Crisis Response Team:** If you or a friend are experiencing food or housing insecurity, or any unforeseen financial crisis, visit [sdsu.edu/ecrt](http://sdsu.edu/ecrt), email [ecrt@sdsu.edu](mailto:ecrt@sdsu.edu), or walk-in to Well-being & Health Promotion on the 3rd floor of Calpulli Center.

**Resources for students:** A complete list of all academic support services--including the Writing Center and Math Learning Center--is available on the Student Affairs' Academic Success website. Counseling and Psychological Services (619-594-5220) offers confidential counseling services by licensed therapists; you can Live Chat with a counselor at [http://go.sdsu.edu/student\\_affairs/cps/therapist-consultation.aspx](http://go.sdsu.edu/student_affairs/cps/therapist-consultation.aspx) between 4:00pm and 10:00pm, or call San Diego Access and Crisis 24-hour Hotline at (888) 724-7240.

**Academic Honesty:** The University adheres to a strict policy prohibiting cheating and plagiarism. Examples of academic dishonesty include but are not limited to:

- copying, in part or in whole, from another's test or other examination;
- obtaining copies of a test, an examination, or other course material without the permission of the instructor;
- collaborating with another or others in work to be presented without the permission of the instructor;
- falsifying records, laboratory work, or other course data;
- submitting work previously presented in another course, if contrary to the rules of the course;
- altering or interfering with grading procedures;
- assisting another student in any of the above;
- using sources verbatim or paraphrasing without giving proper attribution (this can include phrases, sentences, paragraphs and/or pages of work);
- copying and pasting work from an online or offline source directly and calling it your own;
- using information you find from an online or offline source without giving the author credit;
- replacing words or phrases from another source and inserting your own words or phrases.

The California State University system requires instructors to report all instances of academic misconduct to the Center for Student Rights and Responsibilities. Academic dishonesty will result in disciplinary review by the University and may lead to probation, suspension, or expulsion. Instructors may also, at their discretion, penalize student grades on any assignment or assessment discovered to have been produced in an academically dishonest manner.

**Classroom Conduct Standards:** SDSU students are expected to abide by the terms of the Student Conduct Code in classrooms and other instructional settings. Prohibited conduct includes:

- Willful, material and substantial disruption or obstruction of a University-related activity, or any on-campus activity.
- Participating in an activity that substantially and materially disrupts the normal operations of the University, or infringes on the rights of members of the University community.
- Unauthorized recording, dissemination, or publication (including on websites or social media) of lectures or other course materials.
- Conduct that threatens or endangers the health or safety of any person within or related to the University community, including

1. physical abuse, threats, intimidation, or harassment.
2. sexual misconduct.

Violation of these standards will result in referral to appropriate campus authorities.

### Diversity & Inclusion

In our Python “Programming in mathematics” course, we are committed to fostering an inclusive learning environment that recognizes and values the contributions of diverse groups to the development of mathematical and programming knowledge. Our curriculum emphasizes that code, data, algorithms, and computational solutions are often shaped by social, cultural, historical, and political contexts. We encourage students to critically analyze the impact of their technical work on global economic, environmental, and societal systems. Additionally, we strive to equip students with the skills necessary to collaborate effectively in diverse teams, ensuring that they are prepared to contribute meaningfully in a global and multifaceted workforce. This approach aligns with our department's dedication to integrating diversity and inclusion into all aspects of our teaching and learning outcomes.

### Interacting with me

I'll try to respond within 24-48 hours to emails sent to me (sdsu.math340.shen@gmail.com). For quick questions, the turnaround time may be much shorter. (If Canvas is not working or for non-course-related communications, write to me using my work email address: bshen@sdsu.edu.)

Please feel free to request a Zoom meeting, but email is preferred. My regular office hours are Monday, Wednesday, and Friday from 2:30 to 3:30 pm Pacific Time.

### Course Outline

Schedule	Topics	Remarks
<b>Week 1-3</b>	<b>Introduction to Python Programming</b> <ul style="list-style-type: none"> <li>• Overview</li> <li>• IPython, Jupyter, and Google Colab</li> <li>• Demos for General Programming (Curses, Signal Handling, TCP/IP Client &amp; Server, Game, etc.)</li> <li>• Python Shell</li> <li>• Basic syntax and data types</li> <li>• Control structures: Conditional Statements and Loops</li> <li>• Advanced Data Structures</li> <li>• Functions and Modules</li> </ul>	
<b>Week 3</b>	<b>Graphical Representation and Visualization</b> <ul style="list-style-type: none"> <li>• matplotlib package for general plots</li> <li>• seaborn package for statistics plots</li> </ul>	
<b>Week 3-4</b>	<b>Numerical Python (NumPy) Package for Arrays and Vectors</b> <ul style="list-style-type: none"> <li>• In-place arithmetic</li> <li>• Array indexing and array slicing</li> <li>• Vectorization for efficient computations</li> </ul>	
<b>Week 4-5</b>	<b>Error Estimates</b> <ul style="list-style-type: none"> <li>• Triangle inequality</li> <li>• Taylor's remainder theorem</li> <li>• Integral inequality</li> <li>• Relative error</li> <li>• Truncation error</li> <li>• Floating-point numbers represented in machines</li> <li>• Round-off error</li> </ul>	

<b>Week 5-6</b>	<b>Sequences, Series, and Special Functions</b> <ul style="list-style-type: none"> <li>• Fibonacci sequence</li> <li>• <math>3n+1</math> sequence</li> <li>• Prime numbers and the fundamental theorem of arithmetic</li> <li>• Bessel Function</li> </ul>	
<b>Week 7</b>	<b>Symbolic Mathematics in Python (SymPy) for Computational Algebra</b> <ul style="list-style-type: none"> <li>• Numeric Types</li> <li>• Symbolic Differentiation and Integration</li> <li>• Ordinary Differential Equations (ODEs)</li> <li>• Matrix Data Arrangement: Row-major and Column-major</li> <li>• Linear systems of algebraic equations</li> <li>• Eigenvalues and Eigenvectors</li> <li>• Ill-conditioning and condition number</li> </ul>	
<b>Week 8</b>	<b>Introduction to Feedforward Neural Network and Transformer</b> <ul style="list-style-type: none"> <li>• Word embedding and cosine similarity</li> <li>• Backpropagation and chain rule (differentiations)</li> <li>• PyTorch for automatic differentiation (AD)</li> <li>• Residual connection</li> </ul>	
<b>Week 9</b>	<b>Numerical Differentiation and Finite Difference Methods</b> <ul style="list-style-type: none"> <li>• Euler method and finite difference method</li> <li>• General divided differences</li> <li>• General solutions of finite difference equation</li> <li>• Discretization of <math>y' = \lambda y</math> and Fibonacci sequence</li> <li>• Discretization of the Logistic differential equation for the Logistic map</li> </ul>	
<b>Week 10</b>	<b>Difference Equations and Matrix Methods</b> <ul style="list-style-type: none"> <li>• Boundary value problem (BVP)</li> <li>• Sparse matrices</li> <li>• Derivatives using Chebyshev polynomials</li> </ul>	
<b>Week 11</b>	<b>Interpolation and Splines</b> <ul style="list-style-type: none"> <li>• Interpolation techniques</li> <li>• Runge phenomenon</li> <li>• Lagrange interpolation polynomials</li> <li>• Clustered meshes (non-uniform grids)</li> <li>• Linear splines</li> <li>• Cubic splines</li> </ul>	
<b>Week 12</b>	<b>Numerical Integration</b> <ul style="list-style-type: none"> <li>• Numerical quadrature</li> <li>• Trapezoid method</li> <li>• Simpson's method</li> <li>• Adaptive quadrature</li> </ul>	
<b>Week 13 (A)</b>	<b>Root Finding</b> <ul style="list-style-type: none"> <li>• Root finding methods (Newton's method, Secant method)</li> <li>• Rate of convergence</li> </ul>	Subject to the availability of time.
<b>Week 13 (B)</b>	<b>Principal Component Analysis (PCA)</b> <ul style="list-style-type: none"> <li>• Basis vectors and basis functions (e.g., Bessel functions)</li> <li>• Eigenvalues and eigenvectors</li> <li>• Dimensionality reduction</li> </ul>	
<b>Week 13 (C)</b>	<b>SciPy Package</b> <ul style="list-style-type: none"> <li>• Interpolation</li> <li>• Optimization</li> </ul>	

	<ul style="list-style-type: none"> <li>• Integration</li> <li>• Fast Fourier Transform</li> <li>• Signal Processing</li> </ul>	
<b>Week 14</b>	<b>Advanced Topics</b> <ul style="list-style-type: none"> <li>• Object-oriented Programming with Python</li> <li>• Parallel Computing (e.g., MPI4PY, threading module, etc)</li> <li>• ParaView</li> </ul>	<u>Optional(*)</u>
<b>Week 14</b>	<b>Final Review</b> <ul style="list-style-type: none"> <li>• Review of key topics and problem-solving session.</li> </ul>	

\*Optional: these lectures will be presented subject to time availability.