UNM VALENCIA

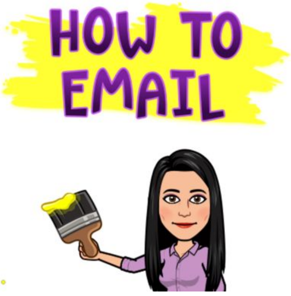
Syllabus-Fall 2024

Title of Course-Section:	**CS 105L-501/502 (Intro. to Computer Programming)**
Name of Department:	Mathematics, Engineering, & Computer Science (MECS)
Instructor:	Andisheh Dadashi, Assistant Prof. of Mathematics (andisheh@unm.edu)
Credit Hours :	3 credit hours
Class Days/Times:	M/W 9 a.m. - 10:15 a.m.
Class Location:	VAAS-127 (Hybrid: Simultaneous Zoom and face-to-face)
Office Location:	VAAS-105
Office Hours:	In-person & Zoom: M 11:50 a.m. - 2:50 p.m. (or by appointment)

Note 1: Zoom links related to this course or office hours can be found on Canvas.
Note 2: The instructor reserves the right to change the syllabus at any point of time during the semester.
Note 3: MECS Division Chair Dr. Ariel Ramirez (aramirez8@unm.edu)

Table of content

Contacting your instructor via Email



When emailing me, in the subject of your email, please write down your course name, course number, and section number. For example, the subject of your email to me should be: **CS 105L-501/502**
You must only contact me with your **UNM e-mail**. Check your **UNM email frequently**. You are responsible for missing any announcement I send via email or posted on Canvas. Failure to identify your message with the class number, and not using your UNM email will result in no response at all.

Instructor Response Time:

- The best way of contacting me will be via email (with proper subject mentioned Above).
- In all cases please, be patient and give me 24 hours to 48 hours to reach back to you.
- I will be available via email during the weekday until 5 p.m.
- I may not be able to respond to any email on Saturday and Sunday.

Get to know your instructor:

Andisheh Dadashi is an Assistant Professor of Mathematics and Computer Science in the Department of Mathematics within the Engineering and Computer Science Division (MECS) at the University of New Mexico-Valencia. She earned her Bachelor's degree in Mathematics and Statistics in Iran, followed by her first Master's degree in Statistical Mathematics in India. Andisheh obtained a second Master's degree in Statistics from the University of New Mexico (UNM) in 2016, marking the beginning of her teaching career as a full-time visiting Lecturer II at UNM-Gallup, where she initiated the implementation of the Critical Technology Studies Program (CTSP) from the main campus. Since then, Andisheh has prepared many student scholars for careers in the Intelligence Community (IC) and other related national security fields.
Andisheh earned a Ph.D. from the Computer Science Department at UNM in May 2024. Her research interests encompass computational biology, genetics, and metabolic networks. Currently, her research is centered on the development of theoretical frameworks, computational methods, and statistical tools aimed at uncovering the mechanisms of rapid polygenic adaptation in response to environmental changes. This involves studying how complex traits in organisms can rapidly adjust through genetic changes, enabling adaption to fluctuating environmental conditions—a vital area of study given the accelerating impacts of climate change on diverse biological systems.

What is Intro. to Computer Programming?

Introduction to Computer Programming is a gentle and fun introduction. Students will use a modern Integrated Development Environment to author small programs in a high-level language that does interesting things. If you are planning on being a CS major and have no previous exposure to computer programming or want a disciplined approach to it, this is the course for you. If you have prior experience, you might want to start with CS152L.

Why Choose Python?

Python has evolved from a humble scripting language into one of the most widely embraced programming languages today. It stands out for several compelling reasons:

1. Readability: Python is renowned for its high-level syntax, making it exceptionally easy to read and write code.

2. Interpreted Language: Python is an interpreted language, which means you can write and execute code without the need for complex compilation processes.

3. Foundational Syntax: Learning the syntax of any language is crucial, and Python is no exception. These rules serve as the building blocks for crafting solutions to problems.

4. Accessibility: Python is open source and free to use for anyone. Additionally, a wealth of free tools and libraries are available to enhance your coding experience.

5. Cross-Platform Compatibility: While some minor issues may arise with file management, Python is mostly cross-platform, allowing you to work seamlessly across different operating systems.

6. Versatility: Python's versatility extends to mobile devices, enabling it to run on phones, which is a notable advantage.

7. Abundant Resources: Python boasts an extensive ecosystem of learning resources and examples, making it one of the most well-supported languages for newcomers and experienced programmers alike.

8. Global Accessibility: When you publish your code using Python, it becomes accessible to anyone worldwide. Python's clarity and simplicity ensure that others can understand and run your code effectively.

In summary, Python's readability, interpretive nature, foundational syntax, accessibility, cross-platform capabilities, mobile compatibility, extensive resources, and global reach make it an outstanding choice for aspiring programmers and seasoned developers alike.

How to be successful in a Programming course?

Success in this course depends on a combination of effort, effective strategies, and a solid understanding of the subject matter. Here are some tips to help you excel in such a course (**What to do**):

1. Attend Classes Regularly: - Make sure to attend all lectures and labs. Consistent attendance helps you understand the material as it's presented, and you won't miss important announcements or assignments.

2. Participate Actively: - Engage in class discussions, ask questions, and seek clarification when you don't understand something. Active participation can help you grasp concepts faster.

3. Read the Course Syllabus: - Understand the course requirements, grading criteria, and important deadlines. This will help you stay organized and plan your study schedule accordingly.

4. Start Early: - Don't procrastinate. Begin assignments and projects as soon as they are assigned. Programming often requires time for trial and error, so starting early allows you to troubleshoot issues.

5. Practice Regularly: - Programming is a skill that improves with practice. Spend extra time coding beyond what's required for your assignments. Experiment with different problems and solutions to reinforce your understanding.

6. Use Online Resources: - The internet is a treasure trove of programming resources. Websites like Stack Overflow, and GitHub can be invaluable for finding solutions to problems and learning from others.

7. Seek Help When Needed: - Don't be afraid to ask for help from your instructor or teaching assistants if you're struggling with a concept or project. They are there to assist you.

8. Understand the Basics: - Ensure you have a solid grasp of the foundational concepts like variables, data types, control structures (if statements, loops), and functions. These will form the building blocks of more advanced programming.

9. Test Your Code: - Always test your code thoroughly before submitting it. Debugging is a crucial skill, and finding and fixing errors is a big part of programming.

10. Study the Documentation: - Get comfortable with reading and understanding documentation for the programming languages and libraries you're using. Documentation is often a programmer's best friend.

11. Stay Inquisitive: - Technology is always evolving. Stay curious and open to new ideas, and tools.

12. Stay Positive and Persistent: - Programming can be challenging, but maintaining a positive attitude and being persistent in your efforts can make a significant difference in your success.

Remember that programming can be challenging, and setbacks are a part of the learning process. Stay persistent, and with dedication and practice, you can excel in this course.

Evaluation/Grading Methods

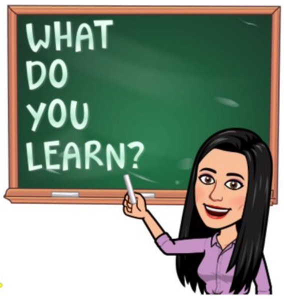
Your final grade in this class is based on the following components:

Progress Credit	10 %
Participation Activities (PA)	25 %
Challenge Activities (CA)	25 %
Programming Challenges or Labs (PC)	25 %
Final Exam	15 %
Overall	100 %

Passing grade is 73% or better. F is a grade lower than 60%
Overall Grades: pluses and minuses may or may not be added to letter grades at the instructor's discretion.
Grades of A+ are not rare and will only be awarded for exceptional work.
In order to pass this course your grade on the final exam must be 70% or better.

Grade	From	To	Grade	From	To	Grade	From	To
A+	98	100	B+	87	89.99	C+	77	79.99
A	93	97.99	B	83	86.99	C	73	76.99
A-	90	92.99	B-	80	82.99	C-	70	72.99

Learning Objectives and Outcomes



Pre-requisites/Co-requisites: If you are planning on being a CS major and have no previous exposure to computer programming or want a disciplined approach to it, this is the course for you. If you have prior experience, you might want to start with CS152L. If you are a non-CS major, your profession might require some computer programming skills, so this is also the course for you.

Course Description: Why do you need this course? Almost every person interacts with a computer program several times a day. All professionals are required to have some knowledge of computers as users. Some professions go even further to ask new employees to be able to write computer programs.

Goals: The objective of the course is an understanding of the relationship between computing and problem-solving. We will be using Python to solve engineering and math problems. The high-level language is Python, the student will learn the basic syntax and rules of Python and will practice it extensively during the term. If you have no previous programming experience, this is the course for you.

By the end of the course, the student will be able to:

1. Design computer solutions to simple problems;

2. Explain how are computer solutions designed;

3. Define basic programming constructs and demonstrate fluency in working with conditional control flow, looping structures, and procedural programming techniques;

4. Write programs to solve simple computer problems in a high-level programming language.

5. Acquire and be able to define and use precisely some terms of computer programming.

6. Establish the foundations of computer programming as the building block for becoming a "good programmer". There is a set of good practices, habits, and skills, related to being a good programmer. Involves characteristics of the product (the program): clear (easy to understand), flexible (easy to modify to make it do something slightly different, or to extend it to add more functionality).

7. Abstract fundamental concepts behind the programming language constructs that will allow the student to learn a new programming language fast, reducing the learning curve of the language.

Technology Requirements

Access to a reliable and fast internet connection is required.

For the course, we use Canvas to navigate through the teaching materials and assignments, but students must also be able to navigate and use other online resources.

Students are required to purchase the online **ZyBook Package** (e-book). ZyBook is the online learning system that accompanies the textbook and includes an e-book. ZyBook is required for the CS 105L course.

How to access Zybook:

- Using your UNM email, sign in or create an account at learn.zybooks.com
- Enter zyBook code: UNMCS105LDadashiFall2024
- Subscribe

Students are required to download and install Python on their computer:

How to download Python:

To learn how to install Python, Anaconda, and other IDE, please, watch the following video that I recorded for you https://youtu.be/XctjwKgChsY
You can use any IDE (Integrated Development Environment) such as Spyder 3, PyCharm CE IDEs, etc for coding in Python.
Spyder IDE is included in Anaconda therefore as long as you download Anaconda you have access to Spyder IDE.
For PyCharm CE IDEs from JetBrain, you can download it from the following website (use your UNM email to create an account for free access): https://www.jetbrains.com/pycharm/download

Course Structure - Book

Book and Package:

The online book will be on learn.zybooks.com. Please, read **above** regarding how to subscribe.
A student's grade is determined by points earned out of 100. The following sections give a breakdown of points; any changes in % or assignments will be based on class needs and communicated early.

- Progress Credit (**10%**)

– Scan the QR Barcode when it is provided to you during class time.
– Answer the questions on the QR code form and submit the form before class ends.
– Work on the Class Zylabs during the class as a group
– There may be Zylab every session or every other session.
– Class Zylabs will be available on ZyBook during the class time
– Class Zylabs must be submitted on ZyBook before the due date
– Credit is given for ongoing progress, incremental achievements, & continuous activity throughout the semester.

- Participation Activities (PA) (**25%**)

– PAs are the section quizzes based on the reading activities from materials in the book
– PAs are due before class
– PAs are not timed
– PAs are posted on ZyBook and must be submitted on ZyBook before the due date

- Challenge Activities (CA) (**25%**)

– CAs are the section quizzes or programming assignments from what you learned during the class
– CAs are due after class
– CAs are not timed
– CAs are posted on ZyBook and must be submitted on ZyBook before the due date

- Programming Challenges (PC) (**25%**)

– PCs are the coding project that is based on the content of each unit
– PCs are due every three to four weeks
– PCs are not timed
– PCs are posted on ZyBook and must be submitted on ZyBook before the due date

- Final (**15%**) exam

– Exam is programming projects or/and multiple-choice questions based on the covered units
– Exam is timed (90 minutes)
– There are 2 trials
– Exam will appear on ZyBook 24 hours before they are due
– Exam will be posted on ZyBook and must be submitted on ZyBook before the due date

Note: All the programs submitted on ZyBook will be automatically reviewed for plagiarism.

Teaching Materials

Q. Where can you find the materials for this class? Canvas and **zyBooks**
Q. Where do you find and submit the assignments? and **zyBooks**
Q. Where do you find your grade? **Canvas** (Gradebook) and **zyBooks**

Course information including this syllabus, and all the necessary materials and links, etc. will be available via Canvas.

QR codes/Attendance/Absence

• - Please download a free QR scanner on your smartphone. You should scan the QR code that I provide for you during the first 5 minutes or the last 5 minutes of class.

• - You are expected to be on time for each class, stay for the entire duration, have the necessary course materials on hand, and participate in the lecture and/or group activities to receive full credit for Progress Credit each day.

• - Even if you miss a class, you are still expected to complete the assignments, but you will lose points for Progress Credit.

Late Assignments

The due dates for the assignments and exam are very firm. Please manage your time wisely in order to prevent any delay. No late assignment is accepted unless in the event of a genuine emergency per the instructor's discretion. Remember that the PAs' assignments are due before each session, and the CAs' assignments are due right after class; therefore, extending these assignments won't make any sense. No early exams will be permitted except in documented emergencies: flight reservations, weddings, vacations, birthdays, non-NCAA sporting events, etc. are not considered emergencies.

Extending assignments for students is not always ideal for several reasons:

1. Maintaining Course Schedule: Extending assignments can disrupt the planned course schedule. Instructors typically design their courses with a specific pace in mind to cover the required material within the available time. Extending assignments may lead to a lag in the curriculum, making it challenging to cover all necessary topics.

2. Developing Time Management Skills: Meeting deadlines is an essential skill for students to develop. By adhering to due dates, students learn how to manage their time effectively and prioritize tasks. Granting extensions too easily can hinder the development of these crucial skills.

3. Fairness and Equity: Granting extensions to some students but not others can create fairness and equity issues. It may lead to resentment among students who met the original deadlines and frustration among those who did not receive extensions.

4. Accountability: Having firm deadlines encourages accountability among students. When assignments have strict due dates, students are more likely to complete them on time, take their work seriously, and strive for quality.

5. Real-World Preparation: Meeting deadlines is a fundamental requirement in most professions. By adhering to assignment due dates in an academic setting, students prepare for the expectations they will encounter in their future careers.

6. Preventing Procrastination: Allowing extensions can encourage procrastination. Students may delay their work with the expectation of getting an extension, leading to last-minute rushes and low-quality work.

7. Maintaining Academic Integrity: Extending deadlines can create opportunities for academic misconduct, such as sharing answers or copying from external sources.

While there are valid reasons to grant extensions in cases of genuine emergencies or extenuating circumstances, it's crucial for instructors to establish clear policies and guidelines for requesting extensions to maintain fairness, consistency, and the educational integrity of the course. In general, extensions should be exceptions rather than the norm to promote responsible time management and a conducive learning environment.

Student Behavior & Collegial Behavior

According to the Code of Conduct as stated in the Policies and Regulations for UNM, student activities that interfere with the rights of others to pursue their education or to conduct their University duties and responsibilities will lead to disciplinary action. This includes any activities that are disruptive to the class and any acts of academic dishonesty. Students are expected to behave in a courteous and respectful manner toward the instructor and their fellow students. Students may be dropped from a class for inappropriate behavior. For more information: Student Code of Conduct
Since we assume you are all adults, we will expect from you, respectful adult behavior. Engaging in disruptive or unruly behavior could result in your being asked to leave, at which time you will be counted absent and a referral will be sent to the Dean of Instruction. Continuing to behave in this way could result in your being dropped from the course. Disruptive or unruly behavior includes but is not limited to:

- texting or talking on your cell phone or Laptop at any time during class,
- continually talking with your neighbor when we are not working on a group activity,
- working on homework from another class,
- refusing to participate in the class activities.
- reading material or watching media not related to this course or at a time that is inappropriate,

Your Responsibility

Time required for This Course: Plan to spend a minimum of 9 to 12 hours per week for this class. There is no guarantee you will pass if you dedicate this amount of time, you still need to learn the material and use your time wisely, but those who pass generally are the ones who spend the time needed to do the work to learn the material.

You are responsible for all material covered in this Syllabus and in class, in assigned readings, and on homework assignments. Not all material on tests will necessarily be covered in class but will be in the assignments. The use of cell phones, headphones, etc. is not permitted in class or exams.

EXPECTATIONS: Students are expected to conduct themselves in a polite, courteous, professional, and collegial manner. Cell phones must be set on silent and be out of sight during class. No food or drink is allowed in the computer labs.

Support!

If you have a documented disability, the "Equal Access Services office" will provide me with a letter outlining your accommodations. I will then discuss the accommodations with you to determine the best learning environment. If you feel that you need accommodations, but have not documented your disability, please contact Cheryl Dilger, the coordinator for Equal Access Services at 925-8910 or cdilger@unm.edu.

If you are struggling in this course, do not be afraid to ask for help!
- Office Hours: See my office hours listed at the beginning of this syllabus."Office Hours" Feel free to come by or log in for online office hours (Info on Canvas), or make an appointment to get help.
- Form study groups: You may work together with other members of our class on the Canvas discussion board.
- Tutoring and Student Services: There are various services provided in our Student Services Department. The Math Center at Valencia campus has free tutoring and open labs. CAPS on the main campus also provides tutoring. For more information please check out the following:

Program Support: CAPS, UNM-Valencia Learning Commons (Tutoring): LRC,
TRIO Student Support Services,
Student Learning Support at the Center for Teaching and Learning., and
Pathways to Articulation and Sustainable Opportunities for Students (PASOS).

Academic Dishonesty

Having academic integrity is paramount to your success in any class. Plagiarism or cheating is not tolerated. Any instance of this will result in a grade of zero for that assignment. Here is the link to the UNM Academic Dishonesty Policy: https://policy.unm.edu/regents-policies/section-4/4-8.html.

The policy states:

Each student is expected to maintain the highest standards of honesty and integrity in academic and professional matters. The University reserves the right to take disciplinary action, up to and including dismissal, against any student who is found guilty of academic dishonesty or who otherwise fails to meet the expected standards. Any student judged to have engaged in academic dishonesty in coursework may receive a reduced or failing grade for the work in question and/or for the course.

Academic Dishonesty is defined as:
"Academic dishonesty" includes, but is not limited to, dishonesty in quizzes, tests, or assignments; claiming credit for work not done or done by others; hindering the academic work of other students; misrepresenting academic or professional qualifications within or without the University; and nondisclosure or misrepresentation in filling out applications or other University records.
Cheating students will be prosecuted according to University guidelines. Students should get acquainted with their rights and responsibilities as explained in the Student Code of Conduct https://grad.unm.edu/aire/academic-integrity.html

UNM Valencia Title IX Representative

Title IX (9) Statement:
In an effort to meet obligations under Title IX, UNM faculty, Teaching Assistants, and Graduate Assistants are considered "responsible employees" by the Department of Education (see pg. 15 - http://www2.ed.gov/about/offices/lis
201404-title-ix.pdf). This designation requires that any report of gender discrimination which includes sexual harassment, sexual misconduct and sexual violence made to a faculty member, TA, or GA must be reported to the Title IX Coordinator at the Office of Equal Opportunity (oeo.unm.edu). For more information on the campus policy regarding sexual misconduct, see: https://policy.unm.edu/university-policies/2000/2740.html

Important Links

Disabilities Policy: Office of Equal Access:
Contact the Office of Equal Access at 925-8560 to schedule an appointment.
https://valencia.unm.edu/students/advisement/equal-access-faqs.html

The Center for Academic Learning:
The Learning Center is open Monday – Friday with evening hours Monday – Thursday To schedule an appointment or for additional information call (505)-925-8907
https://valencia.unm.edu/campus-resources/learning-commons/index.html

UNM Valencia Registrar's Office:

Contact Registration Office by calling 925-8580 http://valencia.unm.edu

UNM Deadlines & Academic Calendar:

UNM Deadlines: https://registrar.unm.edu/semester-deadline-dates/index.html .....*And*....
Academic Calendar: https://hr.unm.edu/calendars

Topics

Topics	In detail
1. Numbers and Strings:	- Data types: Integers, floats, and strings. - Type conversions. - String manipulation methods. - String formatting.
2. Input and Output:	- Reading and writing files. - Standard input and output (stdin, stdout). - Error handling and exception handling.
3. Arithmetic:	- Advanced mathematical operations. - Working with complex numbers. - Numeric libraries like NumPy for scientific computing.
4. Decisions:	- Nested if statements. - Ternary (conditional) operators. - Switch/case statements (not natively available in Python but can be implemented using dictionaries or if-elif chains).
5. Loops:	- While loop - For loop - Nested loops. - Loop control statements (break, continue). - List comprehensions for concise iteration.
6. Functions:	- Advanced function concepts like lambda functions. - Variable scope and global/local variables. - Recursion (calling functions within themselves).
7. Lists:	- List comprehensions. - List methods (append, extend, pop, etc.). - Slicing and indexing. - Multi-dimensional lists and matrices. - Iterating through lists.
8. Sets and Dictionaries:	- Dictionaries operations - Set operations (union, intersection, difference, etc.). - Frozen sets (immutable sets). - Set comprehensions.
9. Strings (Advanced):	- Regular expressions for pattern matching. - String formatting with f-strings. - Working with Unicode characters and encoding/decoding.

The topics listed above cover the foundational aspects of Python programming. Each of these subtopics delves deeper into the corresponding fundamental concepts in Python programming. Depending on your specific goals and projects, you may want to explore one or more of these subtopics to gain a more comprehensive understanding of Python.

Schedule Fall 2024

First day of semester: August 19th & Final Exams: Dec 9th to 14th
Holidays: Labor Day Sep 2nd & Fall break Oct 10th-11th & Thanksgiving Nov 28th-Dec 1st

Week-Day	Schedule subject to change if necessary
W1-D1	Unit 01. Introduction and installation
W1-D2	Unit 01. Computer program and the components of a computer (Part 1)
W2-D1	Unit 01. Computer program and the components of a computer (Part 2)
W2-D2	Unit 01. Python programming and Pseudocode for simple algorithms
W3-D1	No Topic: (Labor Day)
W3-D2	Unit 02. Variables and types
W4-D1	Unit 02. Arithmetic expressions, operators and operands
W4-D2	Unit 02. Strings and Escape Sequences
W5-D1	Unit 02. Input and Output
W5-D2	Unit 03. If Statement
W6-D1	Unit 03. Comparisons of numbers and string
W6-D2	Unit 03. The control flow of a program
W7-D1	Unit 03. Boolean data type
W7-D2	Unit 04. Execution in a loop
W8-D1	Unit 04. While loops
W8-D2	Unit 04. For loops
W9-D1	Unit 04. Carry out complex tasks
W9-D2	Unit 05. Concepts of functions, parameters, and arguments
W10-D1	Unit 05. Returning a value from a function
W10-D2	Unit 05. Principle of stepwise refinement
W11-D1	Unit 05. Local variable and global variable
W11-D2	Unit 05. Importing a toolkit
W12-D1	Unit 05. Recursive function
W12-D2	Unit 06. Lists for collecting values
W13-D1	Unit 06. Built-in operations for lists
W13-D2	Unit 06. Common list algorithms
W14-D1	Unit 06. Tables
W14-D2	Unit 07. Python sets
W15-D1	Unit 07. Built-in operations for sets
W15-D2	No Topic: (Thanksgiving Break)
W16-D1	Unit 07. Python dictionaries
W16-D2	Unit 07. Built-in operations for dictionaries

Due Date Final Exam: 6th Dec, 12:59 pm

Note: Be aware that there are assignments due every few days.
Note: Please find the due dates for PAs, CAs, and in-class labs on ZyBook.