UNM VALENCIA

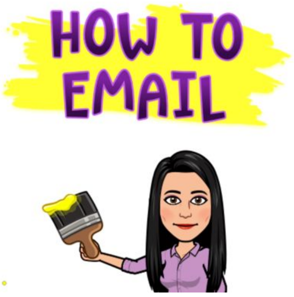
Syllabus-Spring 2024

Title of Course-Section:	**CS 152L-501/502 (Computer Programming Fundamentals)**
Name of Department:	Mathematics, Engineering, & Computer Science (MECS)
Instructor:	Andisheh Dadashi, Assistant Prof. of Mathematics (andisheh@unm.edu)
Credit Hours:	4 credit hours
Class Days/Times:	M/W 10:30 a.m. - 11:45 a.m. & Lab (Fully online) T 9 a.m. - 9:50 a.m.
Class Location:	VAAS-127 (Hybrid: Simultaneous Zoom and face-to-face)
Office Location:	VAAS-105
Office Hours:	In-person & Zoom: TH 7:30 a.m. - 11:30 a.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 152L-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 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 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 student scholars for careers in the Intelligence Community (IC) and other related national security fields.
Currently, Andisheh is a Ph.D. candidate in the UNM Computer Science Department. Her research interests encompass computational biology, genetics, and metabolic networks. At present, 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.
To know **Andisheh** watch this video https://youtu.be/t4ryQfdrSEo

What is Computer Programming Fundamental?

The primary emphasis of this course is to develop fluency in working with conditional control flow, looping structures, and procedural programming techniques. The secondary emphasis is to apply those skills in solving computational problems. The course objectives are understanding relationships between computation, problem-solving, and programming using high-level languages.

Why Choose Java??

Java is a powerful and versatile programming language with numerous advantages that make it a popular choice among developers. Here are some compelling reasons to consider Java for your programming needs:

1. Platform Independence: Java is known for its ”write once, run anywhere” capability. Code written in Java can be executed on different platforms without modification, thanks to the Java Virtual Machine (JVM). This makes Java an excellent choice for cross-platform development.

2. Strongly Typed: Java is a statically-typed language, which means that type errors are caught at compile-time, reducing the chances of runtime errors. This contributes to code stability and maintainability.

3. Community and Ecosystem: Java has a large and active developer community, which means access to a wealth of resources, libraries, and frameworks. Popular frameworks like Spring and Hibernate are widely used in Java development.

4. Security: Java's security features, such as the Java Security Manager and the ability to run code in a sandboxed environment, are essential for developing secure applications.

5. Longevity: Java has been around for decades and continues to evolve. Its long history and continued development make it a stable and future-proof choice for software development.

6. Wide Range of Applications: Java is used in a diverse range of applications, from web and mobile app development (Android apps are primarily written in Java) to backend server applications and embedded systems.

7. High Performance: While Java's execution speed may not be as fast as some lower-level languages, modern Just-In-Time (JIT) compilers have significantly improved its performance, making it suitable for high-performance applications.

In summary, Java's platform independence, reliability, robustness, extensive standard library, strong typing, object-oriented nature, active community, scalability, security features, longevity, application versatility, and performance enhancements through JIT compilation make it a compelling choice for a wide range of programming tasks.

How to be successful in a Programming course?

Success in an introductory computer programming 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, languages, 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 your introductory computer programming course.

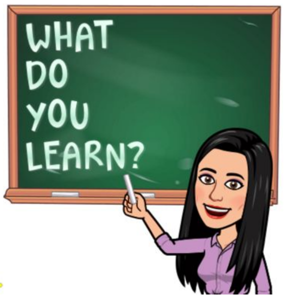
Evaluation/Grading Methods

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

In-class work, attendance, and participation credit	10 %
Participation Activities (PA)	25 %
Challenge Activities (CA)	25 %
Programming Challenges or Labs (PC)	25 %
Final Exam	15 %
Overall	100 %

Passing grade is 70% 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.

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



Pre-requisites/Co-requisites: CS 105L, CS 108L, CS 151L, or ECE 131L.

Course Description The primary emphasis of this course is to develop fluency in working with conditional control flow, looping structures, and procedural programming techniques. The secondary emphasis is to apply those skills in solving computational problems.
CS 152L is a project-based course: students spend many hours writing programs that have a wide range of applications. In past semesters these have included business applications, multimedia manipulations, video games, simulations of complex systems, and scientific models.
CS 152L is currently taught using the Java programming language. While Java is an Object Oriented Programming (OOP)* language and while students in CS 152L will certainly be working with Objects, CS 152L is not a course on OOP. Experienced Java programmers with solid skills in control flow, procedural programming, and computational problem solving should skip CS 152L and take CS 251L (Intermediate Programming). CS 251L is also currently taught in Java and its primary emphasis is on understanding, developing, and applying OOP skills.

Goals: The course objectives are understanding relationships between computation, problem-solving, and programming using high-level languages. This course has several goals. Students who successfully complete the course should have a firm grasp on creating small programs in Java, should be able to solve problems with code, should have a more full idea of what Computer Science as a field is, and most importantly not be afraid to dive into code!

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

1. Develop fluency in working with conditional control flow.

2. Develop fluency in working with looping structures.

3. Develop fluency in working with procedural programming techniques.

4. Design computer solutions to computational problems.

5. Explain how are computer solutions designed.

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

*: Object-oriented programming (OOP) is a computer programming model that organizes software design around data, or objects, rather than functions and logic.

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). This online book will be on Canvas via Inclusive Access (IA). ZyBook is the online learning system that accompanies the textbook and includes an e-book. ZyBook is required for the CS 152L course.

How to access Zybook:

- Using your UNM email, sign in or create an account at learn.zybooks.com
- Enter zyBook code: UNMCS152DadashiSpring2024
- Subscribe

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

How to download JAVA:
Download and install free software from the internet.
Java Standard Edition (SE) Development Kit (JDK) https://www.oracle.com/java/technologies/downloads
Visual Studio Code (recommended editor) https://code.visualstudio.com/Download
IntelliJ IDEA (integrated development environment) https://www.jetbrains.com/idea/download
For IntelliJ IDEA create an account as a student with your UNM email to download the free version of IntelliJ on JetBrains website.
GNU Emacs (alternative editor) https://www.gnu.org/software/emacs/
We may use command-line tools and batch/bash scripts to navigate directories and compile/run java code.

Inclusive Access (IA):
Your course comes with Inclusive Access (IA) which means you will have a **discounted price** of the book using IA on Canvas. On the main page of this course on Canvas, you should find a section named Course Material. After clicking on it you should see a link named RedShelf. After Clicking on RedShelf you must follow the instructions to access the Online Book. Also, you should receive an email that contains instructions for inclusive access to the book via the RedShelf on Canvas. Please, read the email to gain more information regarding the IA.

Book and Package:

The online book will be on Canvas via IA. Please, read **above** regarding Inclusive Access.
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.

• In-class work, attendance, and participation credit (**10%**)
– Scan the QR Barcode when it is provided to you during class time.
– Answer the questions on the QR Barcode and submit the form before class ends.
– Work on the in-class Zylabs during the class as a group
– There may be Zylab every session or every other session.
– in-class Zylabs will be available on ZyBook during the class time
– in-class Zylabs must be submitted on ZyBook before the due date

• 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 3 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.

• Total = **100%**

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 attendance each day.

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

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 lower-quality work.

7. Maintaining Academic Integrity: Extending deadlines can create opportunities for academic misconduct, such as sharing answers or copying from external sources, as students might seek shortcuts when facing time constraints.

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/list/ocr/docs/qa-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. Introduction and Variables:	- Data types in Java (int, double, char, etc.) - Variables, constants, and their scope - Type casting and conversion - Initializing variables - Literals and operators
2. Program Organization:	- Packages and their significance - Import statements - The main() method and program execution flow - Java documentation and comments - Exception handling and error types
3. Control Structures:	- Conditional statements (if, else if, switch) - Looping structures (for, while, do-while) - Nested control structures - Break and continue statements - Using the ‘switch‘ statement effectively
4. Methods (Subroutines):	- Method declaration and syntax - Method parameters and arguments - Method overloading and overriding - Returning values from methods (return statement) - Recursion and recursive methods
5. Classes and Objects:	- Creating and using classes - Object instantiation - Class constructors (default and parameterized) - Class members (fields, methods) - Encapsulation and access modifiers (public, private, protected)
6. Arrays, Searching, and Sorting:	- Declaring and initializing arrays - Array manipulation (adding, removing elements) - Multidimensional arrays - Linear search and binary search algorithms - Sorting algorithms (e.g., Bubble Sort, Quick Sort, Merge Sort) - Time and space complexity analysis of algorithms

Each of these topics is crucial for building a strong foundation in Java programming. Depending on your level of expertise and the specific applications you're interested in, you can delve deeper into any of these areas to gain a more comprehensive understanding of Java.

Additionally, consider exploring topics like object-oriented programming principles, data structures, and design patterns as you progress in your Java programming journey in future.

Schedule Spring 2024

Week-Day	Schedule subject to change if necessary
	First day of semester: Jan 15th & Final Exams: May 6th to 11th Holidays: Martin Luther King Day Jan 15th & Spring break March 10th-17th
W1-D1	(Monday Class No Topic)
W1-D2	Unit 01. Introduction and installation
W2-D1	Unit 01. Computer program and the components of a computer
W2-D2	Unit 01. Java programming
W3-D1	Unit 02. Variables and Assignments
W3-D2	Unit 02. Strings
W4-D1	Unit 02. Input / Output
W4-D2	Unit 02. Streams using Strings
W5-D1	Unit 02. Expressions and Types
W5-D2	Unit 02. Numeric data types
W6-D1	Unit 03. Branches
W6-D2	Unit 03. Switch statements
W7-D1	Unit 03. Evaluation with branches
W7-D2	Unit 04. While loops
W8-D1	Unit 04. For loops
W8-D2	Unit 04. Nested loops
W9-D1	Spring Break No Topic
W9-D2	Spring Break No Topic
W10-D1	Unit 05. Arrays
W10-D2	Unit 05. Enhanced for loop: Arrays
W11-D1	Unit 06. User-Defined Methods
W11-D2	Unit 06. Methods: Common errors
W12-D1	Unit 06. Unit testing (methods)
W12-D2	Unit 06. Java documentation for methods
W13-D1	Unit 07. Objects and Classes
W13-D2	Unit 07. Mutators, accessors, and private helpers
W14-D1	Unit 07. Constructor
W14-D2	Unit 07. Classes and ArrayLists
W15-D1	Unit 08. Exceptions
W15-D2	Unit 09. Memory Management
W16-D1	Unit 10. Recursion
W16-D2	Unit 10. Recursive algorithm: Search

Due Date Final Exam: 3rd May, 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.