EDG 4434: Programming Logic and Algorithmic Design for K-12 Teachers.

3 Credits

COURSE SYLLABUS

Semester: Fall 2026
Course Type: 100% Full Distance Learning
Instructor: Dr. Zafer Unal
Office Hours: Online with Appointment
Phone: 727-873-4803
Email: unal@usf.edu

I. Welcome!

This course focuses on the fundamental principles that underpin all computer programming, specifically designed for K-12 educational settings. We will explore the logical structures that allow us to design correct and efficient solutions to computational problems. You will learn to identify and fix different types of errors, understand the importance of documentation, and compare programming paradigms like procedural and object-oriented programming. The course covers core algorithmic constructs—sequencing, conditionals, iteration, and recursion—and analyzes essential searching and sorting algorithms. We'll also explore the power of propositional logic for designing robust conditions. By the end of this course, you'll have a strong logical foundation for understanding how software works and for teaching computational thinking to your students.

II. University Course Description

This course provides a comprehensive overview of programming logic and algorithmic design for educators. It covers core topics such as error types and debugging, program documentation, procedural and object-oriented programming paradigms, algorithmic constructs (sequence, conditionals, iteration, recursion), searching and sorting algorithms, and propositional logic. The course equips teachers with the foundational logical skills needed to understand program flow and design effective algorithms for K-12 curricula.

III. Course Prerequisites

None

IV. Course Purpose

This course serves as a foundational introduction to the logic that drives computer programs. As coding and computational thinking become integral to modern education, this course equips K-12 teachers with the ability to deconstruct problems, design step-by-step solutions, and understand the core constructs common to all programming languages. The course focuses on developing logical reasoning and analytical skills, making it essential for educators seeking to effectively teach programming concepts and foster a systematic approach to problem-solving in their students.

V. Course Format

This fully online course is structured through weekly modules that combine asynchronous recorded lectures with hands-on technical demonstrations and projects. Students engage through discussion boards for peer interaction and complete practical assignments applying AI tools to educational scenarios. The course offers flexibility through asynchronous learning while maintaining support through individual consultation sessions with the instructor as needed.

VI. Student Learning Outcomes

Upon completion of this course, students will be able to:

- Distinguish between error types (e.g., syntax, runtime, logic) and apply principles of debugging.
- Identify principles, characteristics, and uses of internal and external program documentation.
- Analyze the characteristics and functions of object-oriented and procedural languages.
- Select the appropriate algorithmic sequence, conditional, iteration, and recursive constructs for a given purpose.
- Analyze characteristics and applications of searching (i.e., sequential, binary) and sorting (i.e., selection, insertion, merge) algorithms.
- Analyze the characteristics and applications of propositional logic (e.g., De Morgan's laws).

VII. Course Objectives

Students will:

- Diagnose and categorize errors in provided code snippets and apply systematic debugging strategies to resolve them.
- Create both internal (comments) and external (flowcharts, design documents) documentation for a simple algorithm.
- Compare and contrast procedural and object-oriented programming paradigms, listing advantages and disadvantages of each.
- Design algorithms using pseudocode that correctly implement sequencing, selection, iteration, and recursive constructs to solve given problems.
- Trace the execution of linear search, binary search, selection sort, and merge sort algorithms to predict their behavior and compare their efficiency.
- Apply De Morgan's Laws to simplify and rewrite complex logical conditions used in programming.

VIII. Required Texts and/or Readings and Course Materials

- There are no required text in this course. All of the course materials (videos, readings etc. will be available on the course website with free access. A computer with internet access and capable of running AI applications is required for this course.

IX. Supplementary (Optional) Texts and Materials

NA

X. How to Succeed in this Course

To succeed in this course, students should establish strong study habits from the beginning. This includes completing all weekly readings prior to attempting practical exercises and actively engaging with technical demonstrations. Time management is crucial - start assignments well

before deadlines to allow for troubleshooting and maintain detailed documentation of your technical implementation process. Regular engagement with course materials, peers, and AI tools is essential for building practical skills. Finally, always maintain backup copies of all project work to prevent any potential data loss during technical exercises.

XI. Academic Continuity

As this course is already fully online, any disruptions due to emergencies or severe weather will have minimal impact on course delivery. All course materials, including recorded lectures, assignments, and resources, will remain accessible through Canvas. If USF systems are impacted, backup copies of essential materials will be made available through Microsoft Teams. The instructor will communicate any changes or contingency plans through Canvas announcements and email. While most activities are asynchronous, any scheduled synchronous sessions (like individual consultations) will be rescheduled if disrupted, with alternative times communicated through Canvas.

XII. Communication

Primary communication for this course will be through Canvas messaging and announcements. For urgent matters, you may contact your instructor with the email provided on top. Your instructor typically responds to messages within 24 hours during weekdays and 48 hours on weekends. Virtual office hours are available by appointment through Microsoft Teams. Professional communication etiquette is expected in all interactions - please include your name and course number in email subjects.

XIII. Grading Scale

Grading Scale (%)	
90-100	A
80 - 89	B
70 - 79	C
60 - 69	D
0 - 59	F

XIV. Grade Categories and Weights

The final grade is directly tied to the mastery of each sub-competency. Each assignment will be graded using a specialized rubric designed to assess mastery of its corresponding sub-competency.

Sub-Competency	Assessments	Weight of final Grade
3.1: Distinguish between error types and apply principles of debugging.	Discussion 3.1: Debugging in the Real World, Quiz 3.1: Error Types and Debugging	15%
3.2: Identify principles and uses of internal and external program documentation.	Discussion 3.2: The Value of Documentation, Documentation Assignment 3.2	15%
3.3: Analyze the characteristics of object-oriented and procedural languages.	Discussion 3.3: OOP vs. Procedural, Quiz 3.3: Programming Paradigms	15%

3.4: Select appropriate algorithmic constructs.	Discussion 3.4: Choosing the Right Loop, Algorithm Design Project 3.4	20%
3.5: Analyze characteristics of searching and sorting algorithms.	Discussion 3.5: Real-World Search and Sort, Quiz 3.5: Searching and Sorting Algorithms	20%
3.6: Analyze characteristics of propositional logic.	Discussion 3.6: Logic in Everyday Life, Quiz 3.6: Propositional Logic and De Morgan's Laws	10%
Course Participation	Consistent engagement in weekly modules and discussions	5%
TOTAL		100%

XV. Instructor Feedback Policy & Grade Dissemination

Course modules open every Monday at 12:00 AM and close Sunday at 11:59 PM EST. All assignments must be submitted within this one-week window; late submissions are not accepted given the full week provided for completion. Individual feedback and grades will be posted in Canvas by Wednesday 5:00 PM of the following week. Major assignments like projects and the final portfolio may require additional grading time, which will be communicated in advance. All grades and feedback can be accessed through the Canvas gradebook.

XVI. Course Schedule.

Week	Topics & Subitems	Assessment/Assignment
1	**Programming Logic and Errors (Sub-competency 3.1):** (1) What is Programming Logic? (2) Syntax Errors (3) Runtime Errors (4) Principles of Debugging	**Discussion 3.1: Debugging in the Real World** Share an example of a "bug" you've encountered in any piece of technology or software. Describe how you solved it (or how you would solve it) using a systematic approach.
2	**Debugging and Logic Errors (Sub-competency 3.1):** (1) Logic Errors (2) Tracing Variables (3) Assertion Checks (4) Systematic Troubleshooting	**Quiz 3.1: Error Types and Debugging** Covers the distinguishing characteristics of syntax, runtime, and logic errors, as well as core debugging principles.
3	**Program Documentation (Sub-competency 3.2):** (1) What is Technical Documentation? (2) Internal Documentation (Comments) (3) External Documentation (4) The Program Development Cycle	**Discussion 3.2: The Value of Documentation** Why is documentation important, especially when students are learning to code? Provide an example of a time good (or bad) documentation impacted your ability to understand or use something.
4	**Creating Documentation (Sub-competency 3.2):** (1) Principles of Good Comments (2) Flowcharts and Pseudocode (3) Requirements and	**Documentation Assignment 3.2** Given a short, simple algorithm described in plain English, create both internal documentation (add

	Design Documents (4) Documentation for Maintenance	comments to the pseudocode) and a piece of external documentation (a simple flowchart).
5	**Programming Paradigms (Sub-competency 3.3):** (1) What is a Programming Paradigm? (2) Procedural Programming (3) Object-Oriented Programming (OOP) (4) Key Characteristics of Each	**Discussion 3.3: OOP vs. Procedural** Consider a program for a school library. Would you model it using a procedural or object-oriented approach? Justify your choice by explaining which paradigm's characteristics better fit the problem.
6	**Comparing Paradigms (Sub-competency 3.3):** (1) Languages for Each Paradigm (2) Security and Data Encapsulation (3) Reusability and Modularity (4) Choosing the Right Tool	**Quiz 3.3: Programming Paradigms** Tests the ability to analyze and compare the characteristics, functions, and uses of procedural and object-oriented languages.
7	**Algorithmic Constructs: Sequence & Selection (Sub-competency 3.4):** (1) Sequencing (Top-Down Execution) (2) Selection/Conditionals (IF-THEN-ELSE) (3) Complex Conditions (AND, OR, NOT) (4) Pseudocode Conventions	**Discussion 3.4: Choosing the Right Loop** Describe a simple classroom task that is repetitive. Would you use a FOR loop or a WHILE loop to model it in an algorithm? Explain the difference and why you chose one over the other.
8	**Algorithmic Constructs: Iteration & Recursion (Sub-competency 3.4):** (1) Iteration (FOR, WHILE loops) (2) Choosing Between Loop Types (3) Introduction to Recursion (4) Recursion vs. Iteration	**Algorithm Design Project 3.4** Design an algorithm in pseudocode to solve a given problem (e.g., finding the highest score in a list, calculating a factorial). Your algorithm must correctly use at least one conditional (IF) and one loop (FOR or WHILE).
9	**Searching Algorithms (Sub-competency 3.5):** (1) The Need for Search (2) Sequential/Linear Search (3) Binary Search (4) Comparing Search Efficiency	**Discussion 3.5: Real-World Search and Sort** Give a real-world example of when you use a sequential search in your daily life. Now, give an example of when a binary search would be used (hint: think of a sorted list).
10	**Sorting Algorithms (Sub-competency 3.5):** (1) The Need for Sort (2) Selection Sort (3) Insertion Sort (4) Merge Sort	**Quiz 3.5: Searching and Sorting Algorithms** Tests understanding of the characteristics, execution, and appropriate applications of linear search, binary search, selection sort, insertion sort, and merge sort.
11	**Introduction to Propositional Logic (Sub-competency 3.6):** (1) Propositions and Truth Values (2) Logical Connectives (AND, OR, NOT) (3) Truth Tables (4) Applications in Programming Conditions	**Discussion 3.6: Logic in Everyday Life** Create a simple logical rule for your classroom using AND, OR, and NOT (e.g., "You can have free time IF you have finished your work AND have cleaned your area OR it is a Friday."). Translate this into a logical expression.
12	**De Morgan's Laws (Sub-competency 3.6):** (1) Simplifying Complex Conditions (2) De Morgan's Laws for AND/OR (3) Applying De Morgan's Laws (4) Venn Diagrams and Logic	**Quiz 3.6: Propositional Logic and De Morgan's Laws** Covers the principles of propositional logic and requires the application of De Morgan's Laws to simplify or rewrite logical expressions.

13	**Integration & Application:** (1) Case study analysis (2) Applying all competencies to a complex problem (3) Peer review of final portfolio components	**Final Review Discussion** Analyze a provided, slightly flawed algorithm. Identify the types of errors present, suggest improvements to its logic and documentation, and explain which algorithmic constructs and paradigms are being used.
14	**Synthesis & Portfolio Finalization:** (1) Course synthesis (2) Portfolio assembly and review (3) Future application in K-12 settings	**All Final Assignments Due** Final, polished versions of all major assignments (3.2, 3.4) must be submitted as a cumulative portfolio by the end of Week 14.

* Note: The Schedule is subject to revision

XVII. General Education Statement (undergraduate only – Required if a Gen Ed course)
NA

XVIII. Integration of This Course into Your Academic Experience (Gen Ed courses only)
NA

XIX. Global Citizens Project (only required if a GCP course; must be verbatim)
NA

XX. USF Core Syllabus Policies
USF has a set of central policies related to student recording class sessions, academic integrity and grievances, student accessibility services, academic disruption, religious observances, academic continuity, food insecurity, pregnancy and related conditions, and sexual harassment that **apply to all courses at USF**. Be sure to review these online: usf.edu/provost/faculty-success/resources-policies-forms/core-syllabus-policy-statements.aspx

XXI. Course Policies: Grades
Late Work Policy: Offer specifics about your policy on late work.
Each module provides a full 7-day window for completion, and no late submissions will be accepted after the weekly deadline. To ensure success in this course, it is strongly recommended to begin each module on Monday rather than waiting until the weekend. Starting early allows time to troubleshoot technical issues, engage meaningfully in discussions, and seek clarification if needed. Students who consistently complete work early in the module week typically perform better and experience less stress than those who leave work until the last minute.

Medical Excuses:
If illness prevents coursework completion, students must immediately notify the instructor and submit medical documentation. While this is an online course with no physical attendance requirements, prompt communication regarding any medical issues impacting your ability to complete coursework is essential for arranging accommodations.

Grades of "Incomplete":
For graduate courses: An Incomplete grade ("I") is exceptional and granted at the instructor's discretion only when students are unable to complete course requirements due to illness or other

circumstances beyond their control. The course instructor and student must complete and sign the "I" Grade Contract Form that describes the work to be completed, the date it is due, and the grade the student would earn factoring in a zero for all incomplete assignments. The due date can be negotiated and extended by student/instructor as long as it does not exceed two semesters for undergraduate courses and one semester for graduate courses from the original date grades were due for that course. An "I" grade not cleared within the two semesters for undergraduate courses and one semester for graduate courses (including summer semester) will revert to the grade noted on the contract.

Attendance Policy:
For this online course, regular participation is measured through timely completion of weekly modules and engagement with course materials. Students are expected to log in to the course per week to review content, participate in discussions, and complete assignments.

Campus Free Expression:
It is fundamental to the University of South Florida's mission to support an environment where divergent ideas, theories, and philosophies can be openly exchanged and critically evaluated. Consistent with these principles, this course may involve discussion of ideas that you find uncomfortable, disagreeable, or even offensive. In the instructional setting, ideas are intended to be presented in an objective manner and not as an endorsement of what you should personally believe. "Objective" means that the idea(s) presented can be tested by critical peer review and rigorous debate, and that the idea(s) is supported by credible research. In this course you may be asked to engage with complex ideas and to demonstrate an understanding of the ideas. Understanding and engaging with an idea does not require you to believe it or to agree with it.

Final Examinations Policy:
No exam, non-applicable

XXII. Course Policies: Technology and Media (include sections as applicable to your course)

Canvas:
This course is fully delivered through Canvas Learning Management System. Students must log in regularly to access course materials, assignments, and announcements. Weekly modules open Monday 12:00 AM and close Sunday 11:59 PM EST. All course communication, submission of assignments, and grade distribution will be conducted through Canvas. For technical support, contact USF IT at (813) 974-1222 or help@usf.edu.

XXIII. Course Policies: Student Expectations

Health and Wellness: Example statement supporting student health and wellbeing.

Your health is a priority at the University of South Florida. We encourage members of our community to look out for each other and to reach out for help if someone is in need. If you or someone you know is in distress, please make a referral at www.usf.edu/sos so that the Student Outreach & Support can contact and provide helpful resources to the student in distress. A 24-hour licensed mental healthcare professional, offered through the counseling center, is available by phone at 813-974-2831, option 3. Please remember that asking for help is a sign of strength. In case of emergency, please dial 9-1-1.

Title IX Policy: It is recommended you include the paragraph below verbatim.

Title IX provides federal protections for discrimination based on sex, which includes discrimination based on pregnancy, sexual harassment, and interpersonal violence. In an effort to provide support and equal access, **USF has designated all faculty (TA, Adjunct, etc.) as Responsible Employees, who are required to report any disclosures of sexual harassment, sexual violence, relationship violence or stalking.** The Title IX Office makes every effort, when safe to do so, to reach out and provide resources and accommodations, and to discuss possible options for resolution. Anyone wishing to make a Title IX report or seeking accommodations may do so online, in person, via phone, or email to the Title IX Office. For information about Title IX or for a full list of resources please visit: https://www.usf.edu/title-ix/gethelp/resources.aspx. *If you are unsure what to do, please contact Victim Advocacy – a confidential resource that can review all your options – at 813-974-5756 or va@admin.usf.edu.*

Generative AI: With advancements in AI, tools like GPT-4 can generate human-like text, raising potential issues related to academic integrity and the authenticity of student work. Hence, it's essential to establish clear policies that are communicated to students from the outset of a course. CITL has developed some recommendations that you may consider using in your syllabus here: **CITL Generative AI Syllabus Course Policy Recommendations**

Course Hero / Chegg Policy: Offer specifics about your policy on contract cheating, paper mills, or the use of websites that enable cheating.

Example: The USF Policy on Academic Integrity specifies that students may not use websites that enable cheating, such as by uploading or downloading material for this purpose. This does apply specifically to Chegg.com and CourseHero.com – almost any use of these websites (including uploading proprietary materials) constitutes a violation of the academic integrity policy.

Professionalism Policy:

All course interactions must maintain professional etiquette. Students will communicate respectfully in discussion boards, emails, and group work. Written communications must be clear, appropriate, and maintain a professional tone. Disruptive behavior in online interactions may affect your final grade. For group activities, timely responses and meaningful contributions are required.

Netiquette Guidelines

Professional communication is essential in our online learning environment. When participating in course activities, maintain the same respect and courtesy as in face-to-face interactions. Consider diverse perspectives and remember that written communication lacks verbal cues. Keep messages clear, focused, and constructive. Always proofread for clarity, avoiding all caps and informal language. Complete thoughts on one topic before introducing new ones. Avoid sarcasm and humor that could be misinterpreted. Following these guidelines in Canvas will create a positive learning environment.

End of Semester Student Evaluations:

All classes at USF make use of an online system for students to provide feedback to the University regarding the course. These surveys will be made available at the end of the semester, and the

University will notify you by email when the response window opens. Your participation is highly encouraged and valued.

XXIV. Learning Support and Campus Offices

Academic Accommodations

Students with disabilities are responsible for registering with Student Accessibility Services (SAS) in order to receive academic accommodations. For additional information about academic accommodations and resources, you can visit the SAS website.
SAS website for the Tampa and Sarasota-Manatee campuses.
SAS website for the St. Pete campus.

Academic Support Services

The USF Office of Student Success coordinates and promotes university-wide efforts to enhance undergraduate and graduate student success. For a comprehensive list of academic support services available to all USF students, please visit the Office of Student Success website.

Canvas Technical Support

Include information where students can find technical support.
Example: If you have technical difficulties in Canvas, you can find access to the Canvas guides and video resources in the "Canvas Help" page on the homepage of your Canvas course. You can also contact the help desk by calling 813-974-1222 in Tampa or emailing help@usf.edu.
IT website for the Tampa campus.
IT website for the St. Pete campus.
IT website for the Sarasota-Manatee campus.

Center for Victim Advocacy

Example: The Center for Victim Advocacy empowers survivors of crime, violence, or abuse by promoting the restoration of decision making, by advocating for their rights, and by offering support and resources. Contact information is available online.

Counseling Center

Example: The Counseling Center promotes the wellbeing of the campus community by providing culturally sensitive counseling, consultation, prevention, and training that enhances student academic and personal success. Contact information is available online.
Counseling Center website for the Tampa campus.
Counseling Center website for the St. Pete campus.
Counseling Center website for the Sarasota-Manatee campus.

Tutoring

Example: The Tutoring Hub offers free tutoring in several subjects to USF undergraduates. Appointments are recommended, but not required. For more information, email asctampa@usf.edu.
Tutoring website for the Tampa campus.
Tutoring website for the St. Pete campus.
Tutoring website for the Sarasota-Manatee campus.

Writing Studio

Example: The Writing Studio is a free resource for USF undergraduate and graduate

students. At the Writing Studio, a trained writing consultant will work individually with you, at any point in the writing process from brainstorming to editing. Appointments are recommended, but not required. For more information or to make an appointment, email: writingstudio@usf.edu.
Writing studio website for the Tampa campus.
Writing studio website for the St. Pete campus.
Writing studio website for the Sarasota-Manatee campus.

XXV.Important Dates to Remember

All dates, assignments, and course content are tentative and subject to change at the instructor's discretion. Note: While modules follow a weekly schedule, changes will be announced through Canvas at least one week in advance

For official USF academic deadlines and holidays, refer to the Academic Calendar at http://www.usf.edu/registrar/calendars/