Spring 2021 Programming Languages

Instructor Information

- Name: Dr. Anastasia Angelopoulou
- E-mail: angelopoulou_anastasia@columbusstate.edu
- Office Phone: (706) 507-8047
- School phone: (706) 507-8170
- Office: CCT 433

- Office Hours: TR 11:00am-1:30pm, R 3:00pm-4:30pm and by appointment
- Meeting Time and Place: TR 01:30 PM - 02:45 PM online

Course Information

Course name and number: CPSC 5135 Programming Languages

Credit Hours: 3

Prerequisite: None

Course Description:

This course is designed to introduce students to the fundamental concepts that are the basis for a variety of programming paradigms (e.g., functional, logical, object-oriented, and agent-centric). The course focuses on the principles upon which these different paradigms are based and helping students understand the types of problems each is best suited for. Non-trivial assignments using programming languages representing these paradigms will provide students with a firsthand understanding and appreciation for the strengths and weaknesses of the different approaches. In addition, students will be introduced to some of the concepts underlying the implementation of a language (e.g., lexical, syntax and semantic analysis, symbol tables, binding) through assignments. The course is also intended to give CS students: (1) A broad general education assuring an adequate foundation in science and mathematics relevant to computing; (2) A solid understanding of concepts fundamental to the discipline of computer science; (3) Good analytic, design, and implementation skills required to formulate and solve computing problems; and (4)

The ability to function and communicate effectively as ethically and socially responsible computer science professionals.

Topics for this course include, but are not limited to: Syntax Analysis; Lexical Analysis; Semantic Analysis; Symbol Tables; Static and Dynamic Binding; Control Structures; Functional Programming Languages; Axiomatic, Operational, and Denotational Semantics.

DROP DATE: - (WP)

Required/Recommended Textbooks and Materials

As a reminder, you can purchase your books through the CSU bookstore and can have them shipped directly to you.

REQUIRED TEXTBOOK (FREE ONLINE ACCESS)

- https://opendsa-server.cs.vt.edu/OpenDSA/Books/PL/html/
- http://homepage.divms.uiowa.edu/~slonnegr/plf/Book/

ADDITIONAL OPTIONAL TEXTBOOKS

- Programming Languages: Principles and practice (3rd ed.) 2011 by Louden K.C and Lambert K.A.
- Concepts in Programming Languages by John C. Mitchell
- Concepts of Programming Languages by Richard Sebesta, Addison Wesley publishing, 10th Edition, 2012
- Programming Languages Pragmatics by Michael L. Scott, Morgan Kaufmann, 3rd edition, 2009
- Basics of Compiler Design by Torben Ægidius Mogensen http://web.archive.org/web/20120915222417/http://www.diku.dk/hjemmesider/ansatte/torbenm/Basics/basics_lulu2.pdf

SUPPLEMENTAL MATERIALS

- Additional reading materials will be provided in class and online through the CougarView.

Academic Objectives

1. Students will demonstrate their ability to write non-trivial programs in several languages, including: Lisp, Prolog and C++.
2. Students will be able to define and discuss terms and concepts that are crucial to understanding the semantics of programming languages.
3. Students will be able to analyze a new language, determine the programming paradigm(s) it supports, and discern the types of problems it is best equipped to address.
4. Strategies and Actions used to produce the outcomes:
    1. Lectures and classroom discussions
    2. Individual programming assignments
    3. Team Project
    4. ABET criteria covered: A, B, C, I, J, K
    5. Program objectives covered: 1, 2, 3
    6. Assessment methods: Written exams, project presentations, and programming assignments

Where the ABET criteria are defined as:

A. An ability to apply knowledge of computing and mathematics appropriate to the discipline.
B. An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution.
C. An ability to design, implement and evaluate a computer-based system, process, component, or program to meet desired needs.
D. An ability to function effectively on teams to accomplish a common goal.
E. An understanding of professional, ethical, legal, security, and social issues and responsibilities.
F. An ability to communicate effectively with a range of audiences.
G. An ability to analyze the local and global impact of computing on individuals, organizations, and society.
H. Recognition of the need for, and an ability to engage in, continuing professional development.
I. An ability to use current techniques, skills, and tools necessary for computing practice.
J. An ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices.

K. An ability to apply design and development principles in the construction of software systems of varying complexity.

Course Communication

- Use common sense in writing and sending e-mail.
- Use your official CSU account for all e-mail communication.
- Always identify yourself in the e-mail and let me know what class you attend. If you need clarification on an assignment, ask at least 24 hours before it is due, otherwise you may not get an answer in time to complete the assignment successfully. Read and think about email before sending. Email is a permanent record.

Course Policies

CSU DISABILITY POLICY

If you have a documented disability as described by the Americans with Disabilities Act (ADA) and the Rehabilitation Act of 1973, Section 504, you may be eligible to receive accommodations to assist in programmatic and physical accessibility. We recommend that you contact the Center for Accommodation and Access located in Schuster Student Success Center, room 221, 706-507-8755 as soon as possible. The Center for Accommodation and Access can assist you in formulating a reasonable accommodation plan and in providing support. Course requirements will not be waived but accommodations may be able to assist you to meet the requirements. Technical support may also be available to meet your specific need. For more information go to CSU Center for Accommodation and Access.

COUGARVIEW (D2L BRIGHTSPACE) ACCESSIBILITY INFORMATION

From D2L website: "At D2L we believe learning technologies should never limit learning opportunities. Our accessibility program is tightly integrated with our research and development lifecycle to ensure our tools are standards compliant and easy for people to navigate and understand using the assistive technologies and devices that support their needs... At Desire2Learn we use WAI guidelines, such as the Web Content Accessibility Guidelines 2.0 (WCAG 2.0), Authoring Tool Accessibility Guidelines 2.0 (ATAG 2.0) and

Accessible Rich Internet Applications Suite (WAI-ARIA) to ensure our designs are consistent with international objectives." For more information go to D2L Accessibility.

COURSE ATTENDANCE POLICY

Regular class attendance is required and may be used to determine your involvement in the course in order to satisfy certain Columbus State University reporting requirements. Remember, you are responsible for all the covered material, regardless of your attendance. A student is permitted to accumulate a total of nine hours of absences in a three credit-hour course and may receive a 'WF' grade for excessive absence. Students are expected to account to individual instructors for absences. Refer to the Attendance Policy in the CSU Catalog for more information on class attendance and withdrawal. Anyone absent from a previously announced quiz or test will receive a zero on the quiz or test.

To be permitted to take a final examination at a time other than the date and time published on the Web at http://academics.columbusstate.edu/exams/, students must have permission of the instructor and the dean of the college offering the course. Be aware that non-emergency personal travel is insufficient reason for rescheduling an exam.

HOUSE BILL 280

For information regarding HB 280 (Campus Carry), please refer to House Bill 280. It is the permit holder's responsibility to know and comply with the law.

Time Commitment

The amount of time it will take you to complete the work for the course will depend on many factors, which will vary with each individual. Students can expect to spend anywhere from 8 - 15 hours per week on this course. Consult with the course Calendar and your instructor to be sure you are on schedule, keeping up with the material and submitting assignments and assessments on time.

As a general rule, in this course you will be expected to:

1. make every effort to attend all scheduled class meetings.
2. arrive on time and prepared to listen attentively, taking notes as necessary. You are welcome to use your laptop for taking notes but research shows the old-fashioned way of note taking using pen and paper is actually better.

3. submit your own work on time.
4. check, at least once every weekday, your electronic mail and the CougarView page associated with this course (keep an eye on “Announcements” and “Calendar”).
5. come by and ask me about any questions you may have about the material or any difficulty understanding it, as soon as practical. Since the material builds on itself, later material will not make sense until you understand the previous material.
6. to work hard, keep up, and master the material covered during the semester.

Assignment Requirements

Each problem set or programming assignment will involve analyzing requirements of the problem, designing and representing the solution, or implementing it by coding it using a programming language. Assignments turned in after the due date are considered late. **Late assignments will receive 0 points**. All assignment work must be submitted through the CougarView dropbox.

Programming assignments will be graded not only for the extent to which the programs produce the required results, but also for programming "style". In particular, the programs should be well designed and self-documented with meaningful and informative comments and identifier names. In additional they should also make proper use of the ideas discussed in class. The grading criteria for assignments will vary somewhat between assignments, however the following are basic guidelines which will always apply to programming assignments:

1. Assignments submitted past the deadline will receive a grade of 0.
2. Programs with "compilation" errors will receive a grade of 0.
3. Programs with run-time errors will receive a grade no greater than 75.
4. Programs which run "to completion" but give incorrect results will receive a grade no greater than 90.

A special word of caution about the deadlines for assignments: The assignments may require more time for completion than you might assume at first glance. In fact, some assignments may seem downright easy, but you will find it even easier to underestimate the time required to complete them. Since the penalty for being late is rather severe, you should begin working on each assignment as soon as it is distributed. Remember to allow time for unexpected problems and difficulties; such as minor computer (or network) downtime.

A final note regarding programming assignments: **PLEASE** compile and run your programs immediately before submitting them. Never make a change to the code file(s) and turn it in without compiling and running first. It is *very* easy to make last minute mistakes that introduce syntax or other errors, so checking before submitting can save you an unnecessary grade of 0.

Quizzes

There will be weekly on-line quizzes to be completed and submitted by the time and day indicated in the class schedule.

- Each quiz is composed of the study questions corresponding to the topics on the schedule for that week.
- You will have 6 hours time to complete these quizzes (although you will not need so much time).
- You may use your book, the published notes for this class, internet resources, and course related software (e.g., the C compiler).
- You may not consult any other persons while completing the quizzes. Doing so is not only unethical and dishonorable, it also violates CSU's Academic Integrity guidelines.
- Quizzes started before the deadline but submitted after the deadline will receive a grade of 0.

Feedback and Grading Timeline

For assignments submitted on time, the instructor will be able to grade your submissions and provide feedback within two weeks after the submission deadline unless stated otherwise in the assignment instructions. Feedback will be posted on CougarView under the Grades tab. Please check your settings to make sure that you will receive an alert when you have a new grade posted. The instructor will be monitoring the discussion boards weekly.

Technical Requirements

Students should be able to:

- Compose an email
- Attach a file

- Upload a file
- Download a file
- Save a file to computer or USB device
- Use a Webcam if your course requires Proctor-U or uses web-conferencing.

HARDWARE REQUIREMENTS

GeorgiaView - D2L Help Center, Brightspace compatibility check

SOFTWARE REQUIREMENTS

- An office suite such as Microsoft Office or Open Office
  - Microsoft Office 365 is available free to all CSU students at Microsoft Office 365.
  - OpenOffice can be downloaded for free.
- To open PDF files you might need Acrobat Reader
- Browser Plugins (Pdf files, QuickTime files, Mp4 files) can usually be obtained at the browsers website.
  - Google Chrome
  - Firefox
  - Safari

You can always learn more about CougarVIEW, sometimes referred to as D2L Brightspace, by clicking on the **Resources** navigation menu and selecting **CougarVIEW (D2L) Student Guide"** on the top navigation bar.

If you need technical support or need assistance configuring your computer, you can refer to the link located in the "Technical Resources" widget located on your "My Home" and your "Course Home" pages. If you can not solve your problem after reviewing the knowledge base help pages, you can call the help center 24-7 and talk to a Help Center agent. The number is 1-855-772-0423.

Academic Honesty

(Acknowledgement is hereby given to USG eCore on whose policy this is based).

All students are expected to recognize and uphold standards of intellectual and academic integrity. As a basic and minimum standard of conduct in academic matters that students

be honest and that they submit for credit only the products of their own efforts. Both the ideals of scholarship and the need for fairness require that all dishonest work be rejected as a basis for academic credit. They also require that students refrain from any and all forms of dishonorable or unethical conduct related to their academic work.

In an effort to foster an environment of academic integrity and to prevent academic dishonesty, students are expected to discuss with faculty the expectations regarding course assignments and standards of conduct. In addition, students are encouraged to discuss freely with faculty, academic advisers, and other members of the academic community any questions pertaining to the provisions of this policy.

No cheating in any form will be tolerated. Please be aware that anyone caught cheating or plagiarizing in this class will receive a "0" for the assignment/exam and may receive an F grade for the course. Moreover, a report will be also added to your file and to the BART online system. The second instance of Academic Dishonesty may result in immediate dismissal from the Computer Science programs and expulsion from Columbus State University.

DEFINITIONS AND EXAMPLES

The examples and definitions given below are intended to clarify the standards by which academic honesty and academically honorable conduct are to be judged.

The following list is merely illustrative, and it is not intended to be exhaustive. Moreover, the definitions and examples suggest conditions under which unacceptable behavior of the indicated types normally occurs. However, there may be unusual cases that fall outside these conditions that also will be judged unacceptable by the academic community.

PLAGIARISM

(NOTE: Plagiarism detection systems are often used by CSU faculty members.)

Plagiarism is presenting another person's work as one's own. Plagiarism includes any paraphrasing or summarizing of the works of another person without acknowledgment, including the submitting of another student's work as one's own. Plagiarism frequently involves a failure to acknowledge in the text, notes, or footnotes the quotation of the paragraphs, sentences, or even a few phrases written or spoken by someone else.

The submission of research or completed papers or projects by someone else is plagiarism, as is the unacknowledged use of research sources gathered by someone else when that use is specifically forbidden by the instructor. Failure to indicate the extent and nature of one's reliance on other sources is also a form of plagiarism.

CHEATING ON EXAMINATIONS

Cheating on examinations involves giving or receiving unauthorized help before, during, or after an examination. Examples of unauthorized help include the use of notes, texts, "crib sheets," websites, electronic documents or notes, and computer programs during an examination (unless specifically approved by the instructor), or sharing information with another student during an examination (unless specifically approved by the instructor). Other examples include intentionally allowing another student to view one's own examination and forbidden collaboration before or after an examination.

UNAUTHORIZED COLLABORATION

Submission for academic credit of a work product, developed in substantial collaboration with other person or source but represented as one's own effort, is unauthorized. Seeking and providing such assistance is a violation of academic honesty. However, collaborative work specifically authorized or assigned by an instructor is allowed.

MULTIPLE SUBMISSIONS

It is a violation of academic honesty to submit substantial portions of the same work for credit more than once without the explicit consent of the instructor(s) to whom the material is submitted for additional credit. In cases in which there is a natural development of research or knowledge in a sequence of courses, use of prior work may be desirable, or required. However, the student is responsible for indicating in writing, that the current work submitted for credit is cumulative in nature.

Grading

Students are expected to be fully prepared to discuss the topic(s) each week and complete the assignments each week. Your final grade will be based upon completing the following learning activities and exams. There are three (3) exams scheduled during the semester: two (2) midterm exams and a final exam. All exams are closed book, closed notes, closed neighbor, with no use of laptops, tablets, calculators, phones, or other electronic

devices. The final course average is a composite of the exams, the weekly quizzes, the project, and the programming assignments. The weighting of these items is:

Graded Activities and Exams	
Graded Learning Activities and Exams	**Percentage**
3 Problem Solving/Programming Assignments	20%
2 Midterm Exams	15% each
Team Project	35%
Final Comprehensive Exam	15%
Total Possible	**100%**

Final Grade Calculation	
Percentage Range	**Final Grade**
90-100%	A
80-89%	B
70-79%	C
60-69%	D
59% and below	F

Course Schedule

Spring 2021

The schedule is subject to change. Assignments and project phases will be due on Thursdays, Exams and Project Presentations will take place on Thursdays, unless otherwise specified.

Course Overview			
Week	**Topic**	**Reading/ Video**	**Assignment**

Week #1 Jan 25	Introduction to Programming Languages	Review the Syllabus and Schedule Read Chapter 1 (Introduction) Watch the video "Most Popular Languages 1965-2019" Review the article "How programming languages got their names" Review the website for a detailed list of programming languages with reference links Work on Homework 1 Submit the Teaming Project Form	HW1 and Project released
Week #2 Feb 1	Language Evaluation and Design Criteria	Review the lecture slides Read the "Programming Language Preliminaries" notes Review the practice quiz (answers are also provided)	
Week #3 Feb 8	Lexical Analysis Regular Expressions Finite State Machines	Review the lecture slides Read Sections 1-2 from "Lecture Notes on Regular Languages and Finite Automata" (University of Cambridge) Review the practice questions about regular expressions and finite automata (answers are also provided)	HW1 due
Week #4 Feb 15	Syntax	Review the lecture slides	Project Phase 1 due

	BNF grammars Parse Trees Abstract Syntax Trees	Read the "Context Free Grammars" notes (Chapter 3 of the book "Basics of Compiler Design"). Read Chapter 1: Grammars Read Specifying Syntax Review the practice questions about BNF and Parse Trees	
Week #5 Feb 22	Functional Programming Scheme	Review the lecture slides Read Functional Programming with Scheme Review Midterm Instructions and Topics	HW 2 released **Midterm Exam 1**
Week #6 Mar 1	Semantics Attribute Grammars Annotation Parse Trees	Review the lecture slides Read Attribute Grammars Review the practice problems and solutions	Project Phase 2 due
Week #7 Mar 8	Axiomatic Semantics	Review the lecture slides Read Axiomatic Semantics Review the practice problems and solutions	HW 2 due
Week #8 Mar 15	Axiomatic, Denotational, and Operational Semantics	Review the lecture slides Read Operational Semantics Read Denotational Semantics	Project Phase 3 due

Week #9 Mar 22	Work on project presentation and discuss about upcoming project proposal	Project Presentations	Project Phase 4 due (10-15 min presentation of Phases 1-4)
Week #10 Mar 29	Project Proposal	Review the topics for Midterm 2 Work on practice problems Watch the 10-minute Overview of k Framework Run the K Framework Online	Project Phase 5 due **Midterm Exam 2**
Week #11 Apr 5	Static and Dynamic Binding	Review the lecture slides Watch the videos with Static and Dynamic Scope Examples Review the practice problems and solutions	HW 3 released
Week #12 Apr 12	Symbol Tables	Review the lecture slides Review the practice problems	Project Phase 6 due
Week #13 Apr 19	LaTeX	Review the lecture slides Read LaTeX information on drawing graphics on Overleaf Watch the tutorial for creating LaTeX graphics Review the examples for drawing symbol tables and parse trees	HW 3 due

Week #14 April 26	Project Presentations	Work on final project's deliverables and presentation **Project Presentations**	Project Phase 7 due (Final presentation of Phases 5-7 up to 15 min)
Finals Week (TBD)		**TBD- Study Day (No class)**	
		Final Exam (TBD)	