SER 502 – Emerging Languages and Programming Paradigms

Catalog Description: Emerging programming languages, their design, description, implementation and advanced features. Language paradigms and the resulting implications on execution; run-time systems and storage management; approaches to compilation and execution.

1. **<u>Contact Information</u>**:

   Name: Dr. Ajay Bansal
   Email: ajay.bansal@asu.edu
   Phone: 480-727-1647
   Office: Poly PRLTA 230H

2. **<u>Office Hours</u>**:

   Posted on course site on Canvas

3. **<u>Course Objectives and Expected Learning Outcomes</u>**:

   The course covers the principles of programming and programming languages using a project-based approach that is programming intensive. Various programming language paradigms are used and compared, such as imperative compiled to object-code, and compiled to bytecode, object-oriented, functional, and declarative languages. The course introduces language design (syntax, typing and semantics), implementation approaches (lexical and syntactic analysis), and run-time execution environments. The course involves extensive programming in multiple programming languages and learning to use unfamiliar languages and program development tools. Students develop oral and written communication skills through in-class presentations and writing assignments associated with programming projects. Students may be asked to develop programs in various languages.

   After successfully completing this course, the student can:

   - Communicate, apply and evaluate the tools, activities, and artifacts involved in programming language design, implementation, and program execution.
   - Analyze a problem, identify and effectively communicate candidate languages and associated tools making up the design, development and execution environment for solving the problem.
   - Knowledge of multiple language paradigms, corresponding languages utilizing those paradigms, implementation approaches including runtimes for each, and analysis of their use for problem solving; Especially imperative bytecode compiled, imperative object-code compiled, object-oriented, functional, and declarative.

4. **Grading Policies**:

| Assignments and Projects	50%
Midterm Exam	20%
Final Exam	30%

- All assignments and project must be submitted on the blackboard course site (submissions over e-mail will not be accepted).
- Ample time will be provided to complete assignments and project. The assignments should be turned in by the specified deadline. Late submissions will not be accepted unless prior arrangements have been made with the instructor. It is the students' responsibility to keep a backup of their assignments and projects.
- Students have the right to appeal a grade in writing. Submit your typed appeal with the graded item, stating the reason for your appeal. All appeals must be turned in no later than one week after the item has been returned.

5. **Absence and Make-Up Policies**:

Accommodations can be requested for business or university related travel or illness for more than half the assignment period with appropriate documentation.

Accommodations will be made for religious observances if students notify the instructor at the beginning of the semester concerning those dates. Students who expect to miss class due to officially university-sanctioned activities should inform the instructor early in the semester. Alternative arrangements will generally be made for any examinations and other graded in-class work affected by such absences. The preceding policies are based on ACD 304–04, "Accommodation for Religious Practices" and ACD 304–02, "Missed Classes Due to University-Sanctioned Activities."

6. **Course Materials and Schedule Textbook(s)**:
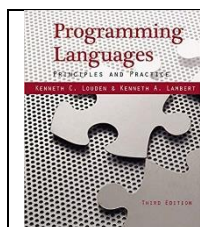
	Recommended (topics are covered from this book): Programming Language Principles and Practice - by KC Louden
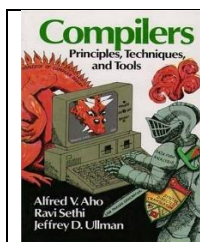
	Useful as a reference: Compilers Principles, Techniques and Tools - by Alfred V. Aho, Ravi Sethi

7. **Course Schedule (Tentative)**:

Week #	*Lecture Content*
Week 1	Class Overview, Introduction to Programming Paradigms
Week 2	Programming Language Design
Week 3	Declarative Programming, Logic Programming
Week 4	Logic Programming (contd).
Week 5	Syntax, Lexical Analysis, Parsing
Week 6	BNFs, Definite Clause Grammar (DCG)
Week 7	Basic Semantics, Data Types, Expressions and Statements
Week 8	Review and Midterm Exam
Week 9	Procedures, Environments, Abstract Data Types and Modules
Week 10	Intermediate Code Generation, Formal Semantics
Week 11	Lambda Calculus, Functional Programming
Week 12	Functional Programming (contd).
Week 13	Constraint Logic Programming
Week 14	Answer Set Programming

8. **Classroom Behavior**:

Cell phones and pagers (must be/or state alternative rule) turned off during class to avoid causing distractions. The use of recording devices is not permitted during class. Any violent or threatening conduct by an ASU student in this class will be reported to the ASU Police Department and the Office of the Dean of Students. The ASU Student Code of Conduct is provided on ASU's website and all students are required to abide by it.

9. **Academic Integrity**:

All students in this class are subject to ASU's Academic Integrity Policy (available at http://provost.asu.edu/academicintegrity) and should acquaint themselves with its content and requirements, including a strict prohibition against plagiarism. All violations will be reported to the Dean's office, who maintain records of all offenses. Students are expected to abide by the FSE Honor Code (http://engineering.asu.edu/integrity/). Punishment can include a record on the student's

transcripts, an E in the course, and/or dismissal from the program. The following exemptions are valid for this course:

- You can discuss the assignment with other students. ***But you are not allowed to copy someone else's code/work***.
- You are encouraged to help other students fix their syntax errors.
- You can discuss the methods and the algorithm with other students. ***But do not write the code (share the code/work) with other students***.

All the code you submit must be yours. A software tool may be used at times to check for similarities between submitted assignments.

10. **Disability Accommodations**:

Suitable accommodations will be made for students having disabilities and students should notify the instructor as early as possible if they will require same. Such students must be registered with the Disability Resource Center and provide documentation to that effect.

11. **Pre-requisites**:

- Fulton engineering graduate student.
- Students are assumed to have the ability to program in an object-oriented programming language, and to have taken a course in data structures and algorithms analysis and design.

12. **Sexual Discrimination**:

Title IX is a federal law that provides that no person be excluded on the basis of sex from participation in, be denied benefits of, or be subjected to discrimination under any education program or activity. Both Title IX and university policy make clear that sexual violence and harassment based on sex is prohibited. An individual who believes they have been subjected to sexual violence or harassed on the basis of sex can seek support, including counseling and academic support, from the university. If you or someone you know has been harassed on the basis of sex or sexually assaulted, you can find information and resources at https://sexualviolenceprevention.asu.edu/faqs.

As a mandated reporter, I am obligated to report any information I become aware of regarding alleged acts of sexual discrimination, including sexual violence and dating violence. ASU Counseling Services, https://eoss.asu.edu/counseling, is available if you wish discuss any concerns confidentially and privately.

Notice: **Any information in this syllabus may be subject to change with reasonable advance notice.**