COURSE DESCRIPTION

Dept., Number	CSci 450	Course Title	Organization of Programming Languages
Semester hours	3	Course Coordinator	Stephen V. Rice, Assistant Professor

Current Catalog Description

History and concepts of programming languages; run-time behavior; formal aspects; language definition; data types and structures; control; and data flow, compilation, and interpretation.

Textbook

Kenneth C. Louden. *Programming Languages: Principles and Practice*, 2nd edition, Brooks/Cole, 2003.

References

Course Outcomes

The Java programming language is the focal point of the introductory course sequence, CSci 111-112-211. CSci 450 surveys a variety of programming languages, examines their features, and gives the student an understanding and perspective beyond the Java-centric view. In the class project, the student researches an unfamiliar programming language and presents an oral and written tutorial of the language. After successfully completing this course, students are able to:

1. read and write common representations of programming language syntax, including syntax trees, BNF, and syntax diagrams
2. describe how static memory and dynamic memory are allocated and reclaimed in typical runtime environments
3. use pass-by-value and pass-by-reference appropriately
4. understand array storage and array-bounds checking
5. describe composition, inheritance and polymorphism
6. read and write a simple functional program in the Scheme language
7. appreciate the rich diversity of programming languages
8. learn how to use an unfamiliar programming language

Relationship between Course Outcomes and Program Outcomes

The course outcomes contribute to the program outcomes as follows: 1 to (a), (j), and (k); 2 to (i); 3 to (b) and (c); 4 through 6 to (i); 7 and 8 to (h).

Prerequisites by Topic

- Basic data structures and algorithms (CSci 112, 211)
- Intermediate programming concepts and skills (CSci 211)
- Fundamental computer architecture concepts (CSci 223)

Major Topics Covered in the Course

- History of programming languages
- Language design principles
- Language translation
- Language syntax
- Language semantics
- Data types and type checking
- Expressions and statements
- Procedures and parameters
- Runtime environments and memory management
- Modules and object-oriented language features
- Introduction to functional programming using Scheme
- Introduction to logic programming using Prolog

Assessment Plan for the Course

A comprehensive, 30-question exam constructed by a faculty committee is administered to each offering of CSci 450. Student performance is analyzed question-by-question to identify needed adjustments in the textbook, lectures, or assignments. Faculty who regularly teach the class participate in the evaluation, in the selection of textbooks, and in formulating a response appropriate to the assessment results.

How Data in the Course are Used to Assess Program Outcomes (unless adequately covered already in the assessment discussion under Criterion 4)

The standard exam administered in CSci 450 (see the previous item) is included in the curriculum–wide outcome assessment described in Chapters 2, 3, and 4 of the Self–Study.

Estimate Curriculum Category Content (Semester hours)

Area	Core	Advanced	Area	Core	Advanced
Algorithms			Software design		
Data structures			Concepts of programming languages		3