CSCI 4342 - ORGANIZATION OF PROGRAMMING LANGUAGES

CREDIT HOURS: 3
PREREQUISITES: CSCI 3302; CSCI 3321 or 3331
GRADE REMINDER: Must have a grade of C or better in each prerequisite course

CATALOG DESCRIPTION

Language definition, structure, data types, control structures, parameter passage, subprogram interface, block structured language. Information binding, data storage and mapping, execution environments, input/output, recursion, multiprocessing.

PURPOSE OF COURSE

To bring to focus the basic and specialized aspects of programming language (PL) constructs and concepts. A variety of PLs are studied and used to illustrate the major PL paradigms. Students will write programs in selected PLs.

EDUCATIONAL OBJECTIVES

Upon successful completion of the course, students should be able to:

1. Describe the significant commonly-accepted criteria for evaluating programming languages (PLs).
2. Identify a large variety of design issues associated with many different PL features.
3. Describe PL design principles.
4. Describe evolutionary progress of the major PLs.
5. Describe the PL paradigms—imperative (procedural), object-oriented, functional (applicative), and logic (declarative).
6. Identify many of the constructs and concepts of a number of PLs in all the paradigms.
7. Work with fellow team members to learn a new PL, write illustrative programs in the assigned PL, develop a hardcopy tutorial for the PL, and make a presentation of the tutorial using multimedia.

COURSE CALENDAR

This course meets for a minimum of 37.5 lecture contact hours during the semester, including the final exam. Students have significant weekly reading and extracurricular assignments. Students are expected to complete weekly homework assignments and quizzes, give a 10-15 minute presentation on a prepared language, complete a major programming project, and 2-3 periodic exams in addition to the final exam. Students are expected to prepare for any class assignments or quizzes over the material covered in class, the reading material, or the extracurricular assignments. Successful completion of these activities requires at a minimum six additional hours of outside of classroom work each week.

CONTENT

<table>
<thead>
<tr>
<th>Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>Preliminary Considerations about PLs</td>
</tr>
<tr>
<td>Evaluation Criteria</td>
</tr>
<tr>
<td>Design Influences</td>
</tr>
</tbody>
</table>
Evolution

PL Constructs and Concepts:
  Syntax and Semantics
  Names, Bindings, Type Checking, and Scopes
  Structured Data Types and Pointers
  Expressions and the Assignment Statement
  Statement-Level Control Structures
  Subprograms
  Data Abstraction
  Concurrency
  Exception Handling

- In Imperative PLs ................................................................. 15
  Traditional (Fortran 77, COBOL); block-structured (Pascal, C, Ada, Fortran 90, C++)

- In Object-Oriented PLs (Smalltalk, C++, Java) ................................................................. 12

- In Functional PLs (LISP, ML, Haskell, SCHEME) ............................................................. 6

- In Logic PLs (Prolog) ......................................................................................... 6

Exams (plus Final) .................................................................................................... 3

TOTAL 45

REFERENCES


