

COP 4020 Programming Languages

Credits and contact hours: 3 credits

Text book, title, author, and year: Concepts of Programming Languages, 10th Edition, Robert W. Sebesta, Addison-Wesley; (January 16, 2012)

Reference materials: Lecture notes, working environments, tooling, and other references will be posted on Blackboard

Specific course information

Catalog description: A comparative study of several higher-level computer languages, including scripting languages such as JavaScript and Perl. Compilers as well as interpreters are discussed. An introduction to formal languages as related to language definition. This course covers the fundamental concepts of programming languages by discussing the design issues of the various language constructs, examining the choices for realizing these constructs in some of the most common languages, and critically comparing design alternatives.

Prerequisites: The prerequisite for this course is Data Structures & Algorithm Analysis. You should have basic understanding different data structures and algorithm, and apply that knowledge in the problem solving by using variety.

Specific goals for the course

The objective of this course is to expose students to:

- Understand a wide range of programming paradigms
- Understand how different programming languages evolved
- Understand the differences in problem domains and language suitability
- Understand the basic features of programming language translation
- Understand implementation techniques for selected language constructs

Specific outcomes of instruction:

- an ability to apply knowledge of mathematics, science, and engineering
- an ability to design a system, component, or process to meet desired needs within realistic constraints such as economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability
- an ability to identify, formulate, and solve engineering problems
- an understanding of professional and ethical responsibility
- an ability to communicate effectively
- an ability to use the techniques, skills, and modern engineering tools necessary for engineering practice

Brief list of topics to be covered:

Course topical outline (subject to change depending on the course progress):

1. The criteria for evaluating languages
2. Primary influences on language design
3. Evolution of most of the important languages
4. Formal methods for describing the syntax and semantics of programming languages
5. Lexical and Syntax Analysis
6. Names, Bindings, and Scopes
7. Data Types
8. Expressions and Assignments Statements
9. Statement-level Control Structures
10. Subprograms and their implementation
11. Abstract Data Types and Encapsulation Constructs
12. Support for Object-Oriented Programming
13. Concurrency
14. Functional Programming Languages