

University of Texas at El Paso
Course Syllabus

COURESE DESCRIPTION

Dept., Number	CS 3360	Course Title	Programming Language Concepts
Approval Date	September 2024	Course Coordinator	Yoonsik Cheon

CATALOG DESCRIPTION

Programming Language Concepts (3-0) Design features of modern programming languages, including flow control mechanisms and data structures; techniques for implementation of these features; practical programming in several representative languages

TEXT BOOK

Robert W. Sebesta. Concepts of Programming Languages, 12th edition, Pearson, 2018.

COURSE OUTCOMES

Level 1: Knowledge and Comprehension:

Level 1 outcomes are those in which the student has been exposed to the terms and concepts at a basic level and can supply basic definitions.

Upon successful completion of this course, students will:

- a. Describe broad trends in the history of the development of programming languages.
- b. Explain the stages of programming language interpretation and compilation.
- c. Understand data and control abstractions of programming languages.
- d. Understand how the attribute grammars describe static semantics.
- e. Describe ways to formally specify the dynamic semantics of small subsets of programming languages, such as expressions and control structures.
- f. Understand code snippets written in a paradigm beyond imperative, object-oriented, and functional, e.g., algebraic, aspect-oriented, logic, or probabilistic languages.
- g. Explain language constructs that promote information hiding and ensure representation independence.

Level 2: Application and Analysis:

Level 2 outcomes are those in which the student can apply the material in familiar situations, e.g., can work a problem of familiar structure with minor changes in the details.

Upon successful completion of this course, students will be able to:

- a. Define the syntax of a small subset of a programming language using BNF.
- b. Compare different approaches to naming, storage bindings, typing, scope, and data types.
- c. Analyze design dimensions of subprograms, including parameter passing methods, sub-programs as parameters, and overloaded subprograms.
- d. Be able to write programs to solve simple problems in a purely functional language.
- e. Be able to write programs to solve simple problems in a scripting language.

Level 3: Synthesis and Evaluation

Level 3 outcomes are those in which the student can apply the material in new situations.

This is the highest level of mastery.

Upon successful completion of this course, students will be able to:

- a. Critically evaluate type system options such as static, dynamic, gradual, and optional and features including type inference, polymorphism, and subtyping of programming languages.
- b. Analyze class inheritance and related code reuse mechanisms in object-oriented programming languages, considering choices such as single vs. multiple inheritance, interface vs. implementation inheritance, and static vs. dynamic dispatch, and mixins.
- c. Assess the utility of advanced expression syntax, including lambda expressions and higher-order functions, in modern programming languages.
- d. Analyze advanced language constructs such as pattern matching, closures, continuations, and concurrency constructs, and critically evaluate their design trade-offs in comparison to traditional programming constructs.
- e. Choose a suitable programming paradigm and language for a given problem or domain.

ABET STUDENT OUTCOMES MAPPING

Course outcomes	Student outcome
2a, 2d-e, 3e	1
3a-d, 3e	2 (ABET 1)
2d-e, 3e	3 (ABET 2)
None	4 (ABET 5)
None	5 (ABET 4)
None	6 (ABET 3)
None	7
None	8
2a, 2d-e	9
3a-d, 3e	10 (ABET 6)

PREREQUISITES BY TOPIC

CS 2302 with a grade of C or better
