# CSEN 171: Design and Implementation of Programming Languages

Fall 2025

| | | | |
|---|---|---|---|
| **Instructor:** | Krishna Ramamoorthy | **Email:** | kkattiyanramamoorthy@scu.edu |
| **Class Meets:** | T-Th 8:30 – 10:10 AM | **Location:** | Alameda Hall 101 |
| **Office Hours:** | T-Th 12:15 – 1:15 PM | **Location:** | Bergin 204 |

**Objectives:** High-level programming language concepts and constructs. Costs of use and implementation of the constructs. Issues and trade-offs in the design and implementation of programming languages. Critical look at several modern high-level programming languages.

**Prerequisites:** CSEN 79 - Object-Oriented Programming and Advanced Data Structures (a grade of C- or better).

## Course Description

This course explores a variety of programming languages and paradigms beyond the introductory level, focusing on their features, design principles, and applications in different domains. Students will gain hands-on experience with multiple programming languages, enabling them to tackle diverse programming challenges. The course aims to foster a deeper understanding of language design, code organization, and effective problem-solving techniques.
The course introduces students to several programming paradigms:

**Imperative Programming**

Imperative programming involves giving explicit step-by-step instructions for the computer to follow.
**C Programming:** Memory management, pointers, data structures, and writing efficient low-level code.

**Logical Programming**

Logical programming languages deal with symbolic reasoning and deduction.
**Prolog:** Rule-based symbolic reasoning and solving logic-based problems.

**Functional Programming**

Functional programming treats computation as the evaluation of mathematical functions
**Haskell:** Lazy evaluation, higher-order functions, type inference, immutability, and functional constructs.

**Object-Oriented Programming**

Object-oriented programming organizes code around objects and classes.
**Principles:** Encapsulation, inheritance, polymorphism, abstraction.
**Java/MATLAB:** Advanced OOP concepts.

## Learning Outcomes

Students will . . .

1. Write programs in several programming languages across different programming language paradigms (e.g., procedural, functional, logical, object-oriented).
2. Specify, infer, and use types in the type system of a programming language.
3. Compare and contrast control structures and mechanisms such as iteration and recursion across different programming languages.
4. Compare and contrast different parameter passing and evaluation strategies.
5. Explain and use different name-value binding (i.e., scoping) implementations.

## Text Books

There are no mandatory textbooks for this course. The textbooks that I will use to cover various topics are listed below.

**Primary References:**

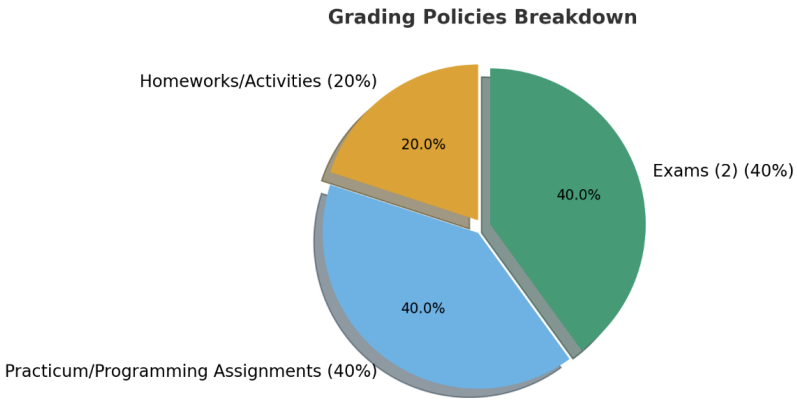
1. Robert W. Sebesta, *Concepts of Programming Languages*, Pearson Education, Inc.
2. Michael L. Scott and Jonathan Aldrich, *Programming Language Pragmatics*, 5th Edition, Morgan Kaufmann.

**Supplemental References:**

1. Stephen G. Kochan, *Programming in C*, Pearson Education, Inc.
2. Miran Lipovaca, *Learn You a Haskell for Great Good!, 1st Edition, No Starch Press.* Online Version available at: `http://learnyouahaskell.com/`
3. Graham Hutton, *Programming in Haskell*, 2nd Edition, Cambridge University Press.
4. Andy H. Register, *A Guide to MATLAB Object-Oriented Programming (Computing and Networks)*, Scitech Publishing.
5. Joyce Farrell, *Java Programming*, 9th Edition, 2018. Cengage Learning.
6. William F. Clocksin and Christopher S. Mellish, *Programming in Prolog*, 5th Edition, Springer.

## Evaluation

Evaluation for this course will be based on multiple measures of performance including, but not limited to: In-class exams, practicum/programming assignments, and participation activities/homework.




**Practicum:** The course will include approximately four practicums to provide hands-on coding experience in the programming paradigms discussed. Each practicum will begin during class hours, allowing students to get started with guidance, and additional time will be provided outside of class to complete and turn-in the work.

**Activities:** The activity problems are designed to help students prepare for exams and to introduce challenging, interview-style questions. Grading will be based on participation: as long as students attend class and make a genuine attempt at the problems, full credit will be awarded.

**Exams:** The exams in this course are designed to assess both the understanding of programming concepts and the ability to implement them in code. Students will be tested on quantitative reasoning and your grasp of the material discussed in class.
There will be **two in-class exams**, both of which are closed-book. These exams emphasize knowledge and problem-solving skills, so you will not encounter multiple-choice or "guess-based" questions.
Please note that not all lecture content is included in the slides. To prepare effectively, you should rely on the following resources: 1. class notes 2. participation activities 3. lecture slides 4. practicum
During exams, collaboration with others is not permitted. The Student may only consult the instructor if clarification is needed. Once a student leaves the exam room, they may not return to continue working on the test.

## Grading Policy

Points are determined by the following categories: In-class activities and homework (20%), Midterm 1 (20%), Midterm 2 (20%), and four practicums / programming assignments (40% = 4 × 10%).

| Score range | Letter Grade | Score range | Letter Grade |
|---|---|---|---|
| $93 \leq$ score | A | $73 \leq$ score $< 77$ | C |
| $90 \leq$ score $< 93$ | A- | $70 \leq$ score $< 73$ | C- |
| $87 \leq$ score $< 90$ | B+ | $67 \leq$ score $< 70$ | D+ |
| $83 \leq$ score $< 87$ | B | $63 \leq$ score $< 67$ | D |
| $80 \leq$ score $< 83$ | B- | $60 \leq$ score $< 63$ | D- |
| $77 \leq$ score $< 80$ | C+ | score $< 60$ | F |

Note: Some problems and short answer questions on exams/assignments are evaluated using coarse grading. It is very difficult to justly and systematically determine that one answer is worth N points, and another is worth N ± x points. Points are assigned based upon broad categories that indicate the degree of mastery:

| A (Excellent) | A- (Good) | B (Mostly Right) | C (On Track) | D (Valiant Effort) | F (Unacceptable) |
|---|---|---|---|---|---|
| 100% | 95% | 85% | 75% | 40% | 0% |

## Late Work Policy

**Exams:** Students must contact the instructor prior to the scheduled exam if they cannot attend. Make-up exams are granted only for approved extenuating circumstances and should not be expected. No make-up exams will be offered once answers have been discussed in class.

**Practicum:** Students will have enough time to work on the practicum. Absolutely no extensions.

**Activities:** Assignments must be submitted by the due date posted in Camino. Late work is generally not accepted, though exceptions may be granted in specific situations. Missed submissions will not be accepted at the end of the semester.

## Course Policies

**Teaching Modality:** All lectures will be held in person and will not be recorded. All exams will be conducted in person and on paper.

**Attendence Requirement:** Students are expected to attend all lectures. Not all lecture materials are included in the slides. Students are encouraged to take notes during the lecture. Attendance at practicum sessions is especially important, as these sessions support the successful completion of assignments.

**Academic Integrity:** The Academic Integrity pledge is an expression of the University's commitment to fostering an understanding of—and commitment to—a culture of integrity at Santa Clara University. The Academic Integrity pledge, which applies to all students, states:

> I am committed to being a person of integrity. I pledge, as a member of the Santa Clara University community, to abide by and uphold the standards of academic integrity contained in the Student Conduct Code.

Academic integrity is part of your intellectual, ethical, and professional development. I expect you to uphold the principles of this pledge for all work in this class. I will clarify expectations on academic integrity—including the use of AI tools such as ChatGPT and course sharing sites—for all assignments and exams in this course. If you have questions about what is appropriate on any assignment, please ask before submitting your work. For more resources on ensuring academic integrity, see this LibGuide on Academic Integrity.

**Plagarism:** Plagiarism detection software will be used to generate similarity reports of your code with your peers, as well as from online sources. Depending on the severity of the similarity, actions will be taken as outlined in the SCU academic integrity policy. The instructor and the grader reserve the right to call upon any student to explain the codes turned in. Failure to comply AND/OR if the student is unable to defend their work/code, the work will be considered as plagiarized; The student will earn no credit and may be reported for academic dishonesty.

**Wellness Statement:** Santa Clara University is committed to fostering a safe, inclusive, and supportive learning environment. Academic integrity, respect, and care for one another are central to our community. Discrimination, harassment, and sexual misconduct are not tolerated; as required by California law (SB 493), faculty must report incidents of sexual harassment or misconduct to the Equal Opportunity and Title IX Office (408-551-3043). Confidential resources include the SCU Wellness Center, Counseling and Psychological Services (CAPS), and professional clergy or counselors. The University also provides accommodations for pregnant or parenting students and for students with documented disabilities through the Office of Accessible Education (OAE). This course affirms people of all gender identities and expressions; please share your name and pronouns with me if they differ from the roster. Above all, remember that your well-being matters—*cura personalis*, concern for the whole person, is at the heart of Jesuit education. Prioritize your health, seek support when needed, and know that many resources, including me, are here to help you succeed.

**Copyright Statement:** Your solutions to exams and other assignments are your own intellectual property. Likewise, my lecture and course materials, including PowerPoint slides, exams, assignments, solutions, etc., are my intellectual property and are protected by copyright laws. You have a right to use these materials for educational purposes while you are an enrolled student in this class. These materials may not be shared, uploaded, or otherwise distributed without my explicit consent. Doing so, even if you do not claim them as your own, is a violation of copyright law and could be subject to disciplinary measures.

**Other policies:** For guidance on academic and classroom expectations, please review the University policiesthat also apply to this course.