

Syllabus

Course Number: CS 390

Course Title: Principles of Programming Languages

Course Description

CS390 Principles of Programming Languages (3). Introduces the constructs upon which contemporary programming languages are based. Students investigate programs written in declarative and imperative programming languages including functional, logic, structured, and object-based approaches.

Prerequisite Courses

CS208 CS Fundamentals **or** CS202 Computational Foundations

AND

CS362 Data Structures **or** CS310 Data Structures

AND

MT320 Discrete Mathematics

Course Overview

Eight primary programming language topics will be covered by this course. They are general programming language principles and evolution, syntax, semantics, design paradigms, construct declaration, typing of data, lambda calculus, and programming languages in practice.

Course Outcomes

Upon completion of this course, learners should be able to:

1. Describe distinguishing characteristics of declarative (functional & logical) and imperative (procedural & object-oriented) programming language paradigms and explain how these characteristics manifest in historic and contemporary programming languages.
2. Evaluate syntactic, semantic, and pragmatic tradeoffs among the various programming paradigms and different programming languages.
3. Summarize the history and continuing evolution of programming languages.
4. Demonstrate different forms of declaration, typing, binding, visibility, scoping, and lifetime management for various programming language constructs (e.g variables, functions, etc.).
5. Use formal systems, including Lambda Calculus, to explain and model various programming language concepts.

6. Develop and analyze programs written in the various programming paradigms.
7. Choose an appropriate programming language solution for a given programming task.

Course Materials

Required Texts:

- Sebesta, S. (2012). *Concepts of Programming Languages* (10th ed.). Boston, MA: Addison-Wesley. ISBN: 978-0-13-607347-5.
- Tate, B. A. (2010). *Seven languages in seven weeks: A pragmatic guide to learning programming languages*. Raleigh, NC: Pragmatic Bookshelf. ISBN: 978-0-13-139531-2.

Required Resources:

Supplemental Weekly Readings (given out in class and online).

Technology Tools:

- SWI-Prolog (free at: <http://www.swi-prolog.org>)
- Haskell (free at: <http://www.haskell.org/haskellwiki/Haskell>)
- Ruby (free at: <http://www.ruby-lang.org/en/downloads/>)

Pre-Assignment:

- Read Chapter 1 of both the Sebesta and Tate textbooks (see below).

Online Format: Sign on to worldclass.regis.edu and become familiar with the course navigation of the Web Curriculum. Complete the assignment above.

Classroom-based Format: Complete assignment above by the first night of class.

Course Assignments and Activities

	Topics	Readings	Activities Assignments and Associated Points*
1	<ul style="list-style-type: none"> Overview & History Language Design Paradigms, Imperative Type Systems Languages in Practice: Ruby 	Sebesta, R. W., (2010): Chapter 1 and §6.1, §6.6, §6.10 – §6.12 Tate, B.A., (2010): §1.0 – §1.1, §2.0 – §2.3 Handouts: (i) Introduction, (ii) Motivational design examples, (iii) Paradigms, (iv) Motivation type examples, (v) Types Systems: Part I (vi) Programming Guidelines	Participation in Discussions (10% for entire course) Assignment #1 (10%) Due: <i>Classroom:</i> Start of Wk 2 class <i>Online:</i> End of Wk 1
2	<ul style="list-style-type: none"> Syntax Language Design: Object-Oriented Lambda Calculus Syntax Languages in Practice: Ruby 	Sebesta: §3.1-:§3.4, §4.1-:§4.5 §12.1-:§12.2 & §12.9 Tate: §2.3 – §2.5. Handouts: (i) Syntax, (ii) Attribute Grammar, (iii) Lambda Calculus I	Assignment #2 (10%) Due: <i>Classroom:</i> Start of Wk 3 class <i>Online:</i> End of Wk 2
3	<ul style="list-style-type: none"> Semantics: Overview & Operational Language Design: Declarative & Logical Lambda Calculus Semantics Languages in Practice: Prolog 	Sebesta: §3.5 & Chapter 16 Tate: §4.1 – §4.2 Handouts: (i) Semantics, (ii) Lambda Part II (iii) Logical Programming (iv) CLP(R)	Assignment #3 (10%) Due: <i>Classroom:</i> Start of Wk 4 class <i>Online:</i> End of Wk 3
4	<ul style="list-style-type: none"> Declarations Language Design Scripting 	Sebesta: Chapter 5 & §2.18 Handouts (i) Declarations (ii) Scripting Languages	Midterm Exam (15%) Due: <i>Classroom:</i> Take Home <i>Online:</i> End of Wk 4

5	<ul style="list-style-type: none"> Semantics Denotational Languages in Practice: Prolog 	Sebesta: §3.5 Tate: §4.3 – §4.5 Handouts: (i) Denotational Semantics (ii) Semantic Algebra (iii) Examples	Assignment #4 (10%) Due: <i>Classroom:</i> Start of Wk 6 class <i>Online:</i> End of Wk 5
6	<ul style="list-style-type: none"> Language Design Functional Lambda Calculus Encoding & Applied Languages in Practice: Haskell 	Sebesta: Chpt. 15 Tate: §8.1-:§8.2 Handouts: (i) Funtional Programming (ii) Lambda Encoding (iii) Applied Lambda Calculus	Assignment #5 (10%) Due: <i>Classroom:</i> Start of Wk 7 class <i>Online:</i> End of Wk 6
7	<ul style="list-style-type: none"> Type Systems: Formal System Lambda Calculus Typed Calculus Languages in Practice: Haskell 	Sebesta: §6.10-:§6.12 Chapter 11 Tate: §8.3-:§8.5 Handout: (i) Typed Lambda Calculus	Assignment #6 (10%) Due: <i>Classroom:</i> Start of Wk 8 class <i>Online:</i> End of Wk 7
8	<ul style="list-style-type: none"> Language Design** <ul style="list-style-type: none"> design criteria abstract data type object-oriented, concurrency, exceptions 	Reading** Sebesta: §1.6 Sebesta: §11.1 - §11.4 Sebesta: §12.1 - §12.2, §12.9 Sebesta: §13.1- §13.2 Sebesta: §14.1, §14.4 - §14.6 Handouts: (i) Design Criteria	Final Exam (15%) Due: <i>Classroom:</i> In class <i>Online:</i> End of Wk 8
Total			100%

***Note to Classroom sections only:** Exact dates for reading assignments and homework assignments may change and will be included in your faculty's syllabus, handed out the first night of class.

** These language design topics are also introduced and discussed in-depth in other courses.

Student Evaluation Summary

Assignment	Value (percent of overall course grade)
Weekly Participation in Discussions	10%
Programming Assignments (6)	60%
Midterm	15%
Final exam	<u>15%</u>
TOTAL	100 %

Course Policies and Procedures

Exams

There will be a midterm exam and a final exam. Exams may be timed; they might be open book, open notes, closed book, and/or closed notes. They might be in-class or take home. Exam questions will be cumulative, taken from reading assignments and class presentations and focus primarily on principles and concepts, as opposed to weekly programming assignments.

Late Assignments Policy

If you have not negotiated with the facilitator, assignments turned in late will be graded, then reduced by 3% per day. Assignments **will not** be accepted later than one week after due date. Week 8 assignment may only be 3 days late.

Adding this course during the Drop/Add Period

If you added this course during the drop/add period, after class began on Monday, you are responsible for *immediately* notifying the instructor that you joined the course late. None of the course due dates will be extended for you. Even if a due date already passed when you added the course, late points will still be deducted.

Plagiarism

Plagiarism includes submitting code or anything else that was obtained from any other person, publication, or any internet web source. **All work submitted in CS390 must be your own.**

In cases of suspected cheating or plagiarism, the instructor will discuss the matter with the student(s) involved. The instructor reserves the right to question any student orally or in writing about any assignment, and to use the evaluation of the student's understanding of the assignment and of the submitted solution as evidence of cheating. All cheating incidents will be reported to the Computer Science department, and may also be reported to the Academic Integrity Board for further action.

CC&IS Grading Scale

Letter Grade	Percentage	Grade Point
A	93 to 100	4.00
A-	90 to less than 93	3.67
B+	87 to less than 90	3.33
B	83 to less than 87	3.00
B-	80 to less than 83	2.67
C+	77 to less than 80	2.33
C	73 to less than 77	2.00
C-	70 to less than 73	1.67
D+	67 to less than 70	1.33
D	63 to less than 67	1.00
D-	60 to less than 63	.67

F	Less than 60	0
---	--------------	---

Additional information about grading can be found in the latest edition of the University Catalog, available at <http://www.regis.edu/Academics/Course%20Catalog.aspx>

CC&IS Policies and Procedures

Each of the following CC&IS Policies & Procedures is incorporated here by reference. Students are expected to review this information each term, and agree to the policies and procedures as identified here and specified in the latest edition of the University Catalog, available at <http://www.regis.edu/Academics/Course%20Catalog.aspx> or at the link provided.

- The CC&IS Academic Integrity Policy.
- The Student Honor Code and Student Standards of Conduct.
- Incomplete Grade Policy, Pass / No Pass Grades, Grade Reports.
- The Information Privacy policy and FERPA. For more information regarding FERPA, visit the [U.S. Department of Education](#)
- The HIPAA policies for protected health information. The complete Regis University HIPAA Privacy & Security policy can be found here: <http://www.regis.edu/About-Regis-University/University-Offices-and-Services/Auxiliary-Business/HIPAA.aspx>.
- The Human Subjects Institutional Review Board (IRB) procedures. More information about the IRB and its processes can be found here: <http://regis.edu/Academics/Academic-Grants/Proposals/Regis-Information/IRB.aspx>.

The CC&IS Policies & Procedures Syllabus Addendum summarizes additional important policies including, Diversity, Equal Access, Disability Services, and Attendance & Participation that apply to every course offered by the College of Computer & Information Sciences at Regis University.

A copy of the CC&IS Policies & Procedures Syllabus Addendum can be found here: <https://in2.regis.edu/sites/ccis/policies/Repository/CCIS%20Syllabus%20Addendum.docx>.