

Syllabus

Course Number: CS 210

Course Title: Introduction to Programming

Course Description:

CS 210. INTRODUCTION TO PROGRAMMING (3). An entry-level course in which students practice software development using elementary selection, looping, method, string, list, array, and object constructs implemented in a modern programming language.

Prerequisite Courses:

None

Course Overview

CS210 is an introductory computer-programming course, which will cover object-oriented programming concepts. The control structures covered in this course form the basis of virtually all structured and object-oriented programming languages.

In this course, the student is introduced to programs that use constant and variables of primitive data types, as well as objects, input/output processing, sequential, selection, and iterative control structures, and classes and methods for modular programming. The student is also introduced to the fundamentals of software engineering methodology.

Ultimately, programming is a skill that requires the ability to translate designs that solve real world problems into programs. The only way to succeed is to practice this skill. Students may require a *significant* amount of time each week to complete the programming assignments.

If you are not willing or able to spend the necessary time, please reconsider whether this is the correct time to attend this class.

Course Outcomes:

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

1. Explain the importance of algorithms in the problem-solving process and create algorithms to solve simple problems using pseudo-code and flowcharting.
2. Compare and contrast the representation and storage of primitive type variables and objects. Determine the scope of a variable within a program.
3. Design, implement, test, and debug programs that use each of the following fundamental programming constructs: basic computation; simple input/output; sequential, conditional,

and iterative control structures; and classes and methods. Choose appropriate constructs for each programming task.

4. Correctly reason about the control flow in a program.
5. Apply the techniques of decomposition to break a program into smaller pieces.
6. Design and implement simple Classes containing data and method members. Use them to instantiate Objects within OOP language programs.
7. Use arrays within programs.
8. Compare and contrast the passing of parameters that are primitive types vs. parameters that are reference types.
9. Describe strategies that are useful in debugging. Use the scientific method and other strategies to identify program deficiencies.
10. Analyze and explain the behavior of programs that use the fundamental programming constructs introduced in this course.

Course Materials:

Required Text:

Adams, J. (2015). *Alice 3 in Action with Java*. Boston, MA: Cengage Learning. ISBN-13: 978-1-133-58918-1 and ISBN-10: 1-133-58918-9.

Technology Tools:

1. A PC-compatible computer system running a version of the Windows operating system, and administrator rights to install new software.
2. The Alice Programming Environment (version 3)
3. The Java programming environment: Java Development Kit (JDK) with NetBeans

Instructions for downloading Alice and JDK are included in the online course shell.

As with most of Regis learning activities, using various software applications to accomplish assignments requires students to exercise a great deal of responsibility for learning how to successfully operate the software applications.

Pre-Assignment:

Complete the following tasks:

Students will read the first week's assigned reading in the textbook (listed in the Course Assignments grid on the next page) before the day of class. Be prepared to ask questions on unclear areas and to respond to questions about information in the assigned reading.

Online Format: Sign on to WorldClass and become familiar with the course navigation of the Web Curriculum.

Course Assignments and Activities*:

Week	Content Topics	Textbook Readings	Activities Assignments and Associated Points
1	1: Computer Science and Programming 2: Hardware and Software 3: Introduction to Alice	Chap 1 , all Chap 2 , sec 2.6 – 2.6.3 (3D)	Pre-Course Survey and Participation in Discussions 10% for entire course Alice Prog Assn 1 – 2.6% Java Prog Assn 1 – 1.6%
2	4: Alice Methods 5: Alice Variables and Expressions 6: Introduction to Java	Chap 2 , sec 2.1 – 2.5 Chap 3 , sec 3.1 (3.1.1 – 3.1.4) Chap 7 , all	Participation in Discussions Alice Prog Assn 2 – 2.6% Java Prog Assn 2 ≈ 3.7%
3	7: More Alice Methods and Variables 8: Java Types and Expressions 9: Java class Methods	Chap 3 , sec 3.2 – 3.5 Chap 8 , sec 8.1 – 8.2.3, 8.3 – 8.4.4 (i.e. skip 8.2.4 for now) Chap 9 , sec 9.1 – 9.2.6	Participation in Discussions Alice Prog Assn 3 – 2.6% Java Prog Assn 3 ≈ 4.9%
4	10: Java Objects and Instance Methods	Chap 9 , 9.3 – 9.5	Participation in Discussions Java Prog Assn 4 ≈ 5.3% Midterm Exam – 18%
5	11: Decision Flow Control (Alice & Java)	Chap 4 , sec 4.1 – 4.2 (Alice) Chap 8 , sec 8.2.4 (boolean) Chap 10 , sec 10.1.1, 10.2.1 – 10.2.2 (Java)	Participation in Discussions Alice Prog Assn 4 – 2.6% Java Prog Assn 5 ≈ 5.3%
6	12: Loop Flow Control (Alice & Java) Last week to withdraw	Chap 4 , sec 4.3 – 4.7 (Alice) Chap 10 , sec 10.1.2, 10.3 – 10.5 (Java)	Participation in Discussions Alice Prog Assn 5 – 2.6% Java Prog Assn 6 ≈ 6.6%
7	13: Files and Exceptions <i>Optional:</i> 14: Arrays in Alice	Chap 11 , all (Java) <i>Optional:</i> Chap 5 , all (Alice)	Participation in Discussions Java Prog Assn 7 ≈ 6.6%
8	15: Arrays in Java <i>Optional:</i> 16: 2D arrays and ArrayLists 17: More OOP Concepts	Chap 12 , sec 12.1.1 – 12.1.4 <i>Optional:</i> Chap 12 , sec 12.1.5 and 12.4 Chap 13 , all	Participation in Discussions Java Prog Assn 8 ≈ 7% Final Exam – 18%
Total			100%

**Note to Classroom sections only:* Exact dates for reading assignments and programming assignments may differ from the above grid. Your instructor's syllabus, handed out the first night of class, will indicate any changes.

Summary of Assignments and Percentage Weight towards course grade

Assignment	Value (percent of overall course grade)
Alice Programming Assignments (5 assns) Assn 1 - 5 at 2.6% each Total for Alice Programming Assignments	13 %
Java Programming Assignments (8 assns) Assn 1 at 1.6% Assn 2 at 3.69% Assn 3 at 4.92% Assn 4 - 5 at 5.33% each Assn 6 - 7 at 6.56% each Assn 8 at 6.97% Total for all Java Programming Assignments	41 %
Midterm Exam	18 %
Final Exam	18 %
Survey/Participation	10 %
Course Total	100 %

Programming Assignments

Each programming assignment will involve writing programs that implement the concepts discussed in the online content, textbook and class (for classroom sections).

Late Assignment Policy for Programming Assignments

Late programming assignments will be graded and then 2% will be deducted for each day the assignment is late, **up to 5 days late**.

No programming assignment will be accepted more than 5 days after the official due date.

Therefore, any programming assignment turned in more than **5 days** late will be given a grade of **zero**, and no feedback will be given.

Exams

There will be a midterm exam and a final exam. Exam questions will be cumulative, taken from reading assignments and course content. **Exams will not be accepted late.**

Participation

Class participation/effort is important because we can all learn from each other. Your participation points can make a difference in the final grade. Participation means:

1. a. Present in class every session (classroom)
b. Present in the forum every week (online)
2. a. Effectively responds to questions from the facilitator (classroom)
b. Regularly checks forum and posts all required items by the deadlines (online)
3. Interacts/replies to other students in classroom/forum discussions.

Online Sections:

See your instructor's syllabus for Online Weekly Discussion points distribution and Rubrics that will be used for grading discussion questions and test cases.

Course Policies and Procedures

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.

Repeating the course

If you are repeating this course (due to a previous withdraw or low grade), you are responsible for ***immediately*** notifying the instructor. Course assignments that you submitted when you last took the course cannot be repeated -- you will be required to complete alternate assignments.

Plagiarism

Plagiarism includes submitting code or anything else that was obtained from another person, a publication, or any internet web source.

Working together on CS210 assignments is not permitted. All work submitted in CS210 must be your own. So while you can discuss the assignments with others, you should NEVER show another student your code!

*** If you receive ANY help from another student, a website, or any other outside source, you must ***cite*** the help in the comments of your code. ***

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](http://www.ed.gov).
- 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>.