

SECTION A-IV: NEW COURSE
Department of Computer and Information

Date of departmental approval: April 12, 2016

Effective date: Fall, 2017

First offering: Spring, 2018

CISC 3115 Introduction To Modern Programming Techniques

4 hours; 4 credits

Bulletin Description

A second course in programming. Programming techniques emphasizing reliability, maintainability, and reusability. Multi-file programs. Abstract data types. Objects, classes, and object-oriented design. Test suites, test drivers, and testing strategies; debugging, assertions, and an introduction to formal techniques. Recursion, event-driven programming and threads, GUI programming, and simple network programming (Not open to students who are enrolled in or have completed Computer and Information Science 3130 [22].)

Prerequisite: CISC 1115 or 1170

Contact hours: 4

Frequency of Offering: every semester

Projected enrollment: 10 sections of 30 students annually

Clearances: None

Rationale: This course is part of a shift from using the C++ programming language to the Java programming language in the early courses of the computer science major sequence. Even more so than the new CISC 1115, this course, which replaces CISC 3110, takes advantages of Java's simpler model for object-oriented programming (OOP) to allow a more comprehensive treatment of OOP, including inheritance and polymorphism. It also takes advantages of the Java core libraries to introduce students to graphical user interface programming, concurrency, and the use of the collection hierarchy.

Department Goals Addressed by Course:

1. Prepare students for the programming requirements in the computer science, information systems or multimedia majors.
2. Prepare students for occasional software development in other disciplines.

Objectives of Course:

1. Understanding the four principles of object-oriented programming— abstraction, encapsulation, inheritance and polymorphism— as well as the concepts of state and behavior, and their realization in the Java programming language;
2. Facility with Java class definition, including the use of constructors and static members;
3. Familiarity with concurrent and event-driven programming and their use in GUI (graphical user interface) application development;
4. Understanding and facility with both recursion and iteration and their application in Java's collection hierarchy;
5. Acquaintance with the Unix programming environment;
6. Understanding responsibility-driven programming and exception handling.

Outcomes Anticipated for Course: At the completion of this course, students will be able to:

1. Develop Java applications involving an application class hierarchy, a GUI interface, and an efficient use of the Java collection packages.
2. Explain object-oriented programming and provide examples of its use in Java.

Course Outline:

1. Unix and the Java development environment [1 week].
2. Java class definitions (including instance variables, instance methods, constructors, scope, visibility, data access and encapsulation, static variables and methods) [2 weeks].
3. Java exception handling [1 week].
4. GUI programming (applets and/or applications, including basic GUI components and their placement, event-driven programming) [2 weeks].
5. Object-oriented programming (interfaces, inheritance, abstract classes, polymorphism, the class hierarchy) [2 weeks].
6. Recursion, and efficient searching and sorting [2 weeks]
7. Java collection hierarchy (including Vector, Set, Map, Iterator, then enhanced for loop, wrapper classes, autoboxing, generics). [2 weeks]
8. Threads (including concurrency, synchronization, animation) [2 weeks].
9. Final Exam (cumulative)
Total 14 weeks (excluding final exam week)

Method of evaluation:

Exams and quizzes

Final exam

Homework assignments and labs

Method of assessment: The department's general assessment methodology will be applied to this course. This involves selecting a subset of questions from the final exam and assessing the student answers in light of overall departmental learning objectives.

Bibliography:

Java: How to Program (Early Objects), Tenth edition. Paul and Harvey Deitel. Pearson, 2014.
ISBN-13: 9780133807806

Core Java Volume I, Tenth edition. Cay S. Horstmann, Pearson, 2015. ISBN-13: 9780134177304

Java Threads and the Concurrency Utilities, First edition. Jeff Friesen. Apress, 2015. ISBN-13:
9781484217009

The Definitive Guide to Java Swing, Third edition. John Zukowski. Apress, 2006. ISBN-13:
9781430200338

<https://docs.oracle.com/javase/tutorial/>