**Objectives: Course Level**

**Course Description**

Will you be mastered by your machinery (computers specifically) or will you become the master? Have you ever wondered what computer programmers do? This course will help you to see how computing professionals solve problems and you will use the high level programming language, Java, to control what your computer will do. We will work together to master basic constructs of computer algorithms, have a little fun with graphical applications, solve problems in small groups, program some simple games and tools, and gain confidence in controlling the machines that have become ubiquitous in our modern society.

We know that you come in with varied experiences with computing, and that all of you have some experience with programming. We provide many help sources to guide your way through this material and for those with more extensive experience, you will have the opportunity to take on additional challenges. In addition to your textbook, which you will find to be a valuable resource in this course, you will have access to video tutorials, which demonstrate concepts and code development throughout the course, as well as instructor office hours and colleague teaching assistant consultation times.

The official catalog description: Students learn fundamental problem-solving techniques using a modern programming language. This course covers the same material as CS 139, but at an accelerated pace for students with programming experience. Students may not receive credit for both CS 139 and CS 149.

*Prerequisites: A prior programming course or equivalent.*

**Themes:**

1. Object-Oriented Paradigm—The OO paradigm models computing as the collaboration of objects rather than the decomposition of operations.
2. UML—The Universal Modeling Language should be used for diagramming.
3. Coding Practice—Computing professionals are comfortable with code; comfort comes from practice reading, writing, and debugging code.
4. Software Engineering Approach—programming is part of a disciplined development effort that requires thoughtful design and coding, desk-checking, and thorough unit and integration testing.
5. Professional Ethics—Intellectual property is owned by its creators and cannot be appropriated without permission. Even when used with permission, its creator must be cited. Turning in someone else’s intellectual property as one’s own is a violation of professional ethics as well as the JMU Honor Code.
6. Reading Specifications--Precise specification of requirements, designs, and so forth, are basic documents in computing. Students must develop skills in reading specifications carefully and understanding exactly what they state.
7. Professional Conduct -- Computer scientists must become adept at "figuring stuff out".  Passive absorption of "lecture" material is not the way to learn in this fast changing environment.  Students should be prepared to spend time engaging with the material in a variety of ways including but not limited to lab work, code walk throughs with fellow students, search for elegant solutions, and practice with programs outside of the class environment.

**Objectives:** The mission of this course is to make you better problem solvers. While controlling the computer is fun, the way we solve problems for the computer can be applied to many other kinds of problems in life. You will learn how to break a problem down into its component parts, determine the best solution to each of those parts, implement that solution in Java, test and correct the solution, and put the components together to form a whole application.

**More specifically, during this course you will:**

* Understand and use appropriate computer programming terminology and concepts. (Found)
* Develop clear and correct algorithms to solve problems on the computer. (Found, App)
* Read and interpret precise application specifications and develop Java programs from those specifications. (App)
* Use appropriate software testing tools and procedures to thoroughly test programs. (App)
* Apply software engineering principles to the act of developing application code. (Int)
* Perform your tasks ethically, being able to differentiate appropriate collaboration from cheating. (Care)
* Communicate effectively with the professor and colleagues about your programming. (Human)
* Use appropriate help resources to accomplish programming tasks.(Learning how to learn)
* Evaluate your own work for compliance with requirements and course style guide.(Care, App)

**We will do this by:**

* using out of class video tutorials, the textbook, and homework assignments to learn new content, and
* working in teams under instructor guidance to apply that content to computing problems, and
* using lab exercises to practice new techniques and more general algorithms, and
* using periodic exams to test your understanding of the underlying principles, and
* using larger programming projects to reinforce and synthesize what you have learned in the tutorials, class, and labs.

This course meets the CS department objectives of:

* ***Programming*:**Students can develop computer programs that solve specific problems using an object oriented programming language. (E)
* ***Problem Solving Methods*:**Students can apply one or more problem solving methods in defining solution requirements and in designing, coding, testing, and documenting a software solution**.** (I)

These objectives were developed through collaboration with the department and other people that are teaching the course.