



CS 139 – Fall 2011

Welcome to Algorithm Development!

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. Rest assured that we understand that many of you have no programming experience. We provide many help sources to guide your way through this material and for those with experience, you will have the opportunity to explore further the same material or 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.

There are no prerequisites to this class.

Your course instructors

Both professors work together on this course. You will find that we will use the same activities in class according to the similar schedules.

Nancy Harris – Sections 1-3 ISAT/CS 217 540.568.2771 harrisnl with the usual jmu.edu address AIM – harrisnl55 Gmail and chat – csharrisnl Lecture – M/W 9:05-9:55 Labs - T/Th Sect 1 - 9:30 – 10:45 Sect 2 - 11:00 – 12:15 Sect 3 - 2:00 – 3:15 Office hours – To be determined by you	Vinnie Capaccio – Sections 4-5 Lecture – M/W 4:40 – 5:30pm Labs - T/Th Sect 4 - 3:30 – 4:45 Sect 5 - 5:00 – 6:15 Office hours
--	---

We will also have a number of undergraduate consultants who have been where you are right now. They will help during the lab times and will have evening and Sunday consulting hours to help. We will provide a schedule of their availability sometime during the first two weeks of class.

Course 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, at the conclusion of this course you will:

NOTE: Italicized levels will be removed for the students. They are for my own use as I am aligning activities.

- Understand the appropriate terminology and structures used in computer programming...sorry, but there are just some things you need to memorize - *foundation*
- Develop clear and correct algorithms to solve problems on the computer – *application, foundation*
- Read and interpret precise application specifications and develop Java programs from those specifications – *application, foundation*
- Apply software engineering principles to the act of developing application code – *integration, application*
- Perform your tasks ethically - *caring*
- Communicate effectively with the professor and colleagues about your programming - *human dimension, application*
- Use appropriate help resources to accomplish programming tasks – *learning how to learn*
- Evaluate your own work for compliance with requirements and course style guide – *learning how to learn, caring, application*

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.

Course Practices

This class has scheduled 2 lecture and 2 lab days each week. Each student will be assigned to either the Monday or the Wednesday lectures.

In **preparation** for the lecture, students will have an online homework assignment involving a video tutorial, a reading assignment, and homework practice problems. On their assigned **lecture** day, students will work in small groups to solve problems related to the homework. **Lab** days will permit students to practice with the new concepts or techniques and to implement the problem solutions developed during the lecture activity. All course materials will be posted on the course schedule and Blackboard, typically before the class period.

Assessment

(15%) Learning and practice – Learning assessments will help you to know how you are doing with class material. These are generally either not graded (quizzes embedded in tutorials) or graded in a binary fashion (you did it or did not do it). We generally have 2 labs per week and at least one homework or in-class assignment per week. I will drop about 10% of your grades which can be anywhere from 3-5 such assignments.

Evaluation – Evaluative assessments will judge your progress toward reaching course goals. These will be graded on a 100 point system. They include project work (PAs) and exams.

(25%) Programming Assignments (PAs) are longer term projects which synthesize a number of individual skills. PAs require you to solve a new problem and follow precise specifications.

(15%) Midterm 1 (week 5)

(20%) Midterm 2 (week 10)

(25%) Final Exam (Practicum, last week of class. Written, Tuesday, Dec 13, 6-8pm)

The relative percentages of exams are cumulative since the material accumulates through the semester and we want you to be able to practice answering the kinds of questions we ask on exams. Therefore, more weight is given at the end of the course than the beginning

Note: Pursuant to the JMU Honor Code, “Each examination, paper and other written or electronically submitted assignment is submitted pursuant to the Honor Code...” ([from the Honor Code site](#)). In this class, evaluative work must be the student’s own work, with assistance from prescribed help resources only. Work primarily for the student’s own learning, may be collaborative. More detail will be provided with each assignment.

General grading – I use a 10 point scale for final grades. 90% and above is an A, etc. I do award +- grades for the final grade for work at the top and bottom of the scale. You need to earn a C or better to continue on to CS 239. I do not give WP or WF grades except under extraordinary circumstances.

Course Resources

Textbook – Starting Out with Java: From Control Structures through Objects, 4/E

Tony Gaddis, *Haywood Community College* ISBN-10: 0136080200. This textbook will be used extensively

through the semester. Used copies are fine, although you may want to be sure that you get the CD so that you have the appendices and code source for examples in the book.

[Java APIs](#) – Oracle (the vendor of Java) provides an extensive library of tools that you will be able to use in your code. Documentation for these tools is available through this link.

[Java Tutorials](#) – Oracle also provides tutorials for java code. These can supplement your understanding of concepts in this course or to let you explore topics that we may not cover directly in class.

CS 139 Tutorials – These are video tutorials that will be available in Blackboard to help you with new material. As we approach each new topic, these tutorials will introduce you to the material. They will remain available throughout the semester.

[CodeBat](#) – CodeBat is a web site which provides practice problems and feedback. When you don't have a specific assignment to work on and you would like more practice, this site provides a great way to gain more skill. I will post specific examples to go with appropriate lessons.

IDE, Compiler, and other software tools – Links to software tools are available through Blackboard. These are all open source, freely available resources.

JAC Card – Put some flex money on your JAC card to use for printing. Printing from the lab or other printers around campus costs about \$.05/page. There may be times when you need to print from the lab.

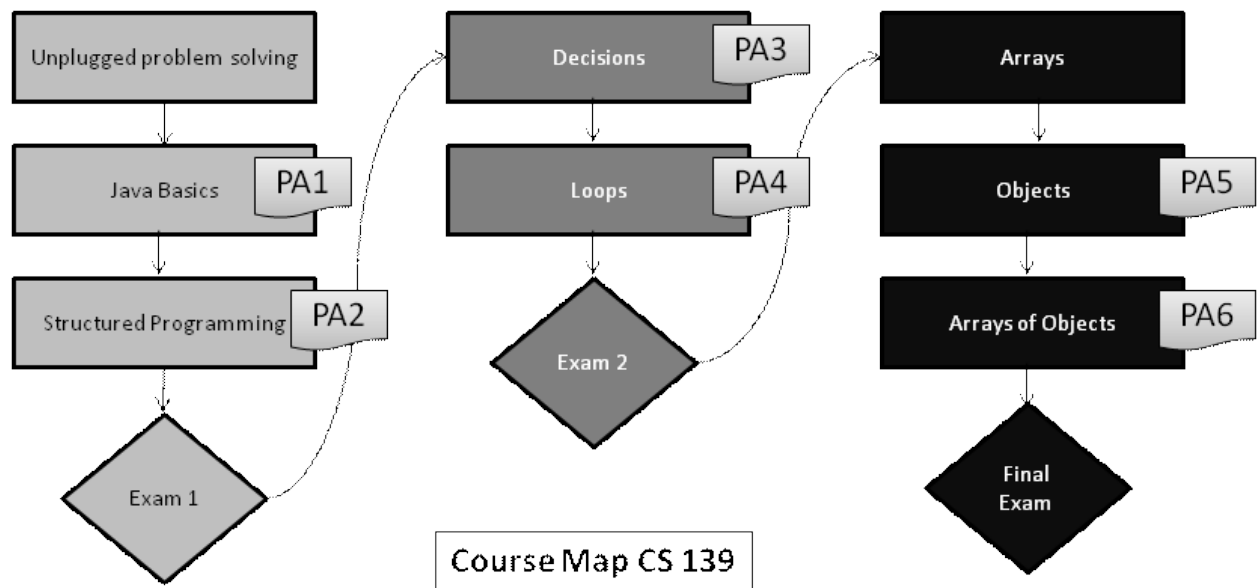
Course Calendar (A separate document)

Student Success

How do you learn to program well? First, it does not come by sitting in a classroom listening to me talk. Think about sports you have played or an instrument or tool that you have learned to use. You need to practice to become better and better. You will be successful in this class if:

- You practice nearly every day (Plan to spend 5-10 hours per week outside of class).
- You study content outside of class so that ...
 - ✓ you come to class every day prepared to work on computing problems with your colleagues, and
 - ✓ you use the labs to actively explore new concepts and ideas, and
 - ✓ you get started early on programming assignments, and
 - ✓ you never forget that whether you are a beginner or experienced programmer, you should ask for help in your learning when needed.

<< an aside not in the final syllabus>> The Course map below is subject to some refinement as I build the schedule. This does accurately reflect the ordering of topics as one builds upon the next.



The fine print:

Students with Disabilities: If you are a student with a documented disability, who will be requesting accommodations in my class, please make sure you are registered with the Office of Disability Services, Wilson Hall, Room 107 (568-6705) and provide me with a copy of your Access Plan letter outlining your accommodations. I will be glad to meet with you privately during my office hours to discuss your special needs. The sooner you can do this, the better I can assist you in meeting your learning goals in this course.

"Excused" absences: Students who are unable to attend class due to JMU sponsored activities (such as sports, band, academic competition, academic field trips, etc) or personal religious observance may request deadline extensions BEFORE the expected absence. I will do my best to accommodate your special circumstances.

School closing for weather or other unexpected circumstances: Watch our Blackboard site for announcements relating to make up work. Generally, if it is a lab day, we will conduct lab using Elluminate. If it is a class day, there will be an alternate individual assignment. Program and exam dates will be adjusted as needed. For online make-up, I will make provision for students who do not have access to the internet or Blackboard. See me on the first class day after the missed class day.