



CS 239 – Advanced Programming – Sections 1 and 2
Spring 2011
Lecture T/Th 11 – 12:15, HHS 1209
Lab M/W 10:10 – 11 and 11:15 – 12:05, ISAT/CS 248

Instructor Information

Ms. Nancy Harris
ISAT/CS 217
540-568-8771
harrisnl@jmu.edu
AIM – harrisnl55; G-chat – csharrisnl
If I am online, please feel free to
contact me.

Office Hours

M 1:30-2:30
T 3-4 and 5-6
W 2:30-3:30
Th 3-4
Other hours by appointment

General Description:

Students use various advanced problem-solving strategies to develop algorithms using classes and objects. Students also learn how to implement and use elementary data structures, including character strings, records, files, stacks and queues. developing clear, concise, and correct algorithms to solve problems on a digital computer.

Prerequisites: CS 139 or equivalent with a grade of "C" or better.

Note: Students planning to continue to CS240 next semester should take CS/Math 227 during this semester. CS 239 and CS 227 are prerequisites to CS 240.

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 through with fellow students, search for elegant solutions, and practice with programs outside of the class environment.

Goals: At the conclusion of this course, successful students will

1. use advanced programming techniques to solve computing problems. These include but are not limited to:
 - o inheritance
 - o polymorphism
 - o enumerated data types
 - o exceptions
 - o abstract classes

- interfaces
 - file I/O
 - recursion
 - complex data structures such as multi-dimensional arrays, ArrayList, HashTable, linked lists.
2. use appropriate design techniques to design software applications before implementation.
 3. use appropriate testing techniques to thoroughly test an application during development.
 4. use the Java APIs and tutorials to learn about new classes and techniques and to supplement textbook information.
 5. understand UML diagrams and their relationship to the design process.
 6. read and understand software specifications to implement code that conforms to the specifications and to coding standards in place for the course.

Textbook and Required Material

Gaddis, Tony (2010) Starting Out with Java, Pearson Education, Boston, MA, 4th edition.

You are welcome to use any edition that you like or you may use a different book. All reading and homework references will be from Gaddis.

Java API's:

<http://java.sun.com/javase/6/docs/api/>

Java Online Tutorial:

<http://java.sun.com/docs/books/tutorial/index.html>

JAC Card:

There may be times in lab that you need to print a worksheet, a completed assignment or other work. You must have your JAC card with you and some money on it for printing. Make sure that you have a couple of dollars on the card each day.

Printing:

In addition to any printing required in the lab, you may need to bring a copy of your completed lab assignment with you to class each lecture day. For those with a laptop, that will be sufficient for sharing in the classroom. For those without, you should print a copy of any worksheets or completed code.

Backup Media:

It will be important that you have a backup device available to use in the lab. While you will get some network space, sometimes the network goes down and you need some way to continue working. Lab machines do not provide a permanent save. When the machine is logged out, your work is lost.

Class Policies and Student Success

Specific policies and items which will insure your success in this class are:

- **Come to class**, come to class, come to class - Students who attend class regularly will keep up with the workload better, will gain insights from talking with classmates or the instructor, and will do better in this class as a result. We use a lot of active learning in class so attendance is mandatory.
- **Workload** - This is a four(4) credit hour class. There is considerable outside work expected. More work will be done at the end of the semester than at the beginning. Be sure to get started on programming assignments early as these may be more difficult than they first appear.
- **Late assignments** - All lab assignments and homework are due at the beginning of class on the day that they are due. No late assignments will be accepted for credit, although I will review late assignments at your request. For programming assignments, late assignments will have a per day penalty associated with them.
- **Exams** - We have two midterms and a final exam. It is expected that all students will take the exams at the normally scheduled time. If an emergency occurs which requires you to be away, let me know as soon as you know about the situation. I may request documentation in order to allow you to take a make-up exam.
- **Honor Code** - Almost every semester I have incidents of students cheating on programming assignments in some fashion. I report ALL incidents of academic dishonesty to the Honor Council. If the violation is severe, I will refer it to the Honor Council for formal resolution. Less severe violations may be handled informally. If you know of cheating in this class, it is your responsibility to let the instructor know as soon as possible. If you are involved in a situation where you are not sure if what you did was right, please see me.
- **Extra Credit** - I generally do not have extra credit opportunities. One exception (and there may be a few others) are students who find bugs in the submit system are given extra points on that assignment only. There are many opportunities in this class for credit.

Student Success - You will be successful in this class if you:

- **Attend** class and lab each day.

- **Engage** with the material...don't passively listen in class and expect to do well on the application assignments.
- **Ask for help** when you need it. We have TA assistance in the evening, and I have plenty of office hours.
- **Get started early** on assignments.
- **Think** about what you are doing...don't just do. In this course, the why is more important than the what.
- **Experiment** with the language and assignments and talk to your colleagues during lab time. Feel free to explore.
- Don't put yourself into a position of providing unauthorized help to another student. Labs are designed for free exploration and that includes working with others. Programming assignments are your way to test what you know and test your problem solving ability. While we may have some programs in which you will be able to work with a partner, most will be individual assignments. If you have a question about what help may be provided, please see me.

If you need additional resources to help you with your learning, let me know. Additional reference books may be placed in the library upon request.

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 or by appointment 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.

Student Responsibility: Students are responsible for adding and dropping courses via e-campus, adhering to the dates posted on the registrar's web site for the semester. I do not give "WP" or "WF" grades to students requesting a drop after the deadline except in extraordinary circumstances.

Religious observance and other planned absences: Students who are unable to attend class due to religious observance, athletic competition, academic competition or academic events may request deadline extensions BEFORE the expected absence. I will do my best to accommodate your special circumstances.

Academic Integrity: Don't put yourself into a position of providing unauthorized help to another student. Labs are designed for free exploration and that includes working with others. Programming assignments are your way to test what you know and test your problem solving ability as well as for me to evaluate your abilities. All programming assignments must reflect individual effort. See the more detailed collaboration policy for what is and is not acceptable with regard to programming assignments posted in Blackboard. If you have a question about what help may be provided, please see me.

Inclement Weather Policies: If JMU closes due to weather or other emergency, (see policy 1309), please access the Blackboard announcements page for this class for information about making up the class. In most situations we will either make up the material at other times in the semester or do the planned work in an online environment.

Grading

All exams are cumulative due to the nature of the material. Each exam will focus on the most recently covered material (since the last exam), but anything covered in the semester is fair game.

Exam 1 – Feb 17	15%
Exam 2 – Mar 31	20%
Final Exam –	
Practical portion – April 27	25%
Written portion - May 4, 6pm – 8pm, ISAT/CS 159	
Programming Assignments - 7	30%
Lab, homework, class participation	10%

NOTE: A student that that does not earn 60% of the points on the final exam will receive a letter grade no higher than a D+ for the course. You must achieve a C or better grade to continue on to CS 240.

I generally grade exams and programming assignments on a 100 point scale. Letter grades correspond to 10 point ranges...90 - 100 = "A", 80 - 89.999 = "B", etc. "F" is used for work below 60%. Labs will be graded based on completeness and your preparation for review in "lecture". Final grades will be based on the algorithm above and may include + / - grades for extremes within a range.