Personnel

Your professor: Dr. Farzana Rahman  
Email: rahma2fx@jmu.edu  
Webpage: www.mscs.mu.edu/~frahman  
Office: ISAT/CS 212

Office Hours:  
Wed/Fri: 10:30 AM - 12:00 PM  
Tue/Thu: 1:00 PM - 2:00 PM  
or by appointment (please email)

Lectures  
Wed/Fri: 9:05 AM - 9:55 AM  
ISAT/CS 236 (classroom)

Lab sections  
Section 1: Tue/Thu, 9:30 AM - 10:45 AM  
Section 2: Tue/Thu, 11:00 AM - 12:15 PM  
ISAT/CS 248 (Linux lab)

Course web page  
wwwww......

Course Description  
Will you be mastered by technology (computers, tablets, phones) or will you become the master? Have you ever wondered what computer scientists do? This course will help you see how computing professionals solve problems. You will use the Java programming language to see what computers can really 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 simple games and tools, and gain confidence in controlling the machines that are everywhere in modern society.

We know that you come in with varied levels of computing experience. Rest assured we understand that many of you have no prior programming knowledge! We provide many help sources to guide your way through this material. For those with more preparation, you
will have the opportunity to explore the same material further or take on other challenges. In addition to your textbook, which you will and to be a valuable resource in this course, you will have access to video tutorials that demonstrate concepts and code development, as well as weekly office hours and teaching assistant consultation times.

And now, the official catalog description: *Students learn fundamental problem-solving techniques using computer software tools that support algorithm development and procedural abstraction to analyze a domain and create reusable software applications.*

**Learning Outcomes**
The mission of this course is to make you a better problem solver. While programming is both fun and useful, the way we formulate solutions for computers 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.
- Develop clear and correct algorithms to solve problems using computation.
- Read and interpret precise application specifications and write code from them.
- Use appropriate software tools and procedures to thoroughly test your programs.
- Apply software engineering principles to the act of developing application code.
- Perform tasks ethically, being able to distinguish appropriate collaboration from cheating.
- Communicate effectively with the professor and colleagues about your programming.
- Use appropriate help resources to accomplish computer programming tasks.
- Evaluate your own work for compliance with requirements and the course style guide.

*We will do this by:*

- Using out-of-class video tutorials, the textbook, and homework to learn new content.
- Working in teams on guided activities to apply that content to computing problems.
- Doing lab exercises to practice new techniques and more general algorithms.
- Taking periodic exams to test your understanding of the underlying principles.
- Completing larger programming projects to reinforce and synthesize what you have learned in the tutorials, classes, and labs.
The Weekly Routine

This course has two lectures and two lab days scheduled each week. Students are expected to attend all classes and actively participate. **Please review the detailed schedule on the course home page as the semester progresses.**

In preparation for the lectures, students will have an online homework assignment involving a video tutorial, textbook readings, and/or practice problems. During class time, students will work in small groups to solve problems related to this preparation.

Lab days will permit students to practice with the new concepts or techniques and to implement the solutions developed during the previous lecture activity. All materials will be posted on the course website or Canvas, typically before the class period.

Undergraduate TAs are assigned to each section. They will help during the lab times and will have evening and Sunday consulting hours as well. We will provide a schedule of their availability during the first two weeks of class.

Grading policies

Your final grade will be based on your performance in the following areas:

- **(20%) Learning and practice**
  Frequent assignments will help you know how you are doing with course material. These are generally either not graded (quizzes embedded in tutorials) or graded in a binary manner (you did it or not, pass/fail). We generally have two lab write-ups per week, and at least one homework or in-class assignment per week.

- **(30%) Programming assignments**
  These “PAs” are longer projects which integrate a number of individual skills. They will require you to solve a new problem and follow precise specifications. We will have six projects during the semester, with deadlines scheduled about every two weeks.

- **(10%) Midterm 1 (Week 5, in class)**
- **(15%) Midterm 2 (Week 10, in class)**
- **(25%) Final Exam (Practicum: last day of class + Written: during finals week)**

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

Letter grades will be assigned on the scale A=90-100, B=80-89, C=70-79, D=60-69, F=0-59, with potential minor adjustments after considering the overall performance of the class and actual distribution of numeric scores. I will use “+” and “−” grades at my discretion. **You need to earn a C or better to continue on to CS 239.** I do not assign WP or WF grades except under extraordinary circumstances.

***Note that you **MUST** pass the final exam to pass the overall course (i.e. at least get a D)***
Late Work Policy
Activities designated as learning or practice must be turned in on time, usually the day they are assigned. I will drop the lowest 10% of your scores in this area to allow for unforeseen circumstances. Project assignments will be due on Thursdays. Except for severe cases (e.g., hospitalization) with proper documentation (e.g., a doctor's note), late PAs will be penalized 10% on Fridays, 20% on Sundays, 35% on Mondays, and 50% on Tuesdays, after which they will no longer be accepted. Unless you have extraordinary circumstances, missed exams CANNOT be made up.

Other policies
During labs, students need to work in pairs. Pair programming is also recommended for Programming Assignments (PAs) starting from PA # 3. For more information on programming in pairs please visit the following link at: http://en.wikipedia.org/wiki/Pair_programming .

Attendance is vital to success in this class. Please refer to the policy on absences in the university bulletin.

Any course related communication will be made to the student’s official JMU email address. It is the responsibility of the student to check his/her emails.

Academic Honesty
“Each examination, paper and other written or electronically submitted assignment is submitted pursuant to the Honor Code” (see http://www.jmu.edu/honor/). In this class, evaluative work must be the student’s own work, with assistance from prescribed help resources only. Academic dishonesty (claiming another person's work as your own) will not be tolerated. Infractions will result in immediate failure of the course, and referral to the Dean's office. We will use Moss (see http://theory.stanford.edu/~aiken/moss/) to detect inappropriate collaboration in programming projects. Limited collaboration is allowed in practice assignments during class. More detail will be provided with each assignment.

Textbook

![JAVA](image)

This textbook will be used extensively through the semester and into next 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.

**Useful Resources**

**Storage** - You will need to have some means of storing your programs and other work in progress. We recommend that you set up an account with DropBox.com or GoogleDocs(Google Drive) or some other cloud facility.

**Java Tutorial:** [http://docs.oracle.com/javase/tutorial/](http://docs.oracle.com/javase/tutorial/)
Oracle (the vendor of Java) provides an official tutorial on the Java language. These lessons can supplement your understanding of course concepts and let you explore topics we may not cover.

**Java APIs:** [http://docs.oracle.com/javase/6/docs/api/](http://docs.oracle.com/javase/6/docs/api/)
The Java runtime comes with an extensive set of libraries that you will be able to use in your programs. Documentation for these libraries is available on Oracle's website.

**CS 139 Tutorials:** [https://blackboard.jmu.edu/](https://blackboard.jmu.edu/)
As we approach each new topic, video tutorials will be available in Blackboard to introduce you to the material. They will remain available throughout the semester.

**Piazza:** [https://piazza.com/jmu/fall2012/cs139](https://piazza.com/jmu/fall2012/cs139)
We will use Piazza to facilitate online Q&A across all sections of CS 139. Instead of emailing your questions directly to me, I encourage you to post them on Piazza for all to benefit. If however you need to schedule an appointment or have a personal inquiry, don’t hesitate to contact me directly.

**Scratch:** [http://scratch.mit.edu/](http://scratch.mit.edu/)
A tool to help learn programming. We will use Scratch in some of the early labs. IDE, Compiler, and other software tools – Links to software tools are available through Canvas. These are all open source, freely available resources.

**CodingBat:** [http://codingbat.com/java](http://codingbat.com/java)
This free site of interactive problems will help you practice Java coding skills. CodingBat allows you to implement small programming tasks (like on an exam) and gives you immediate feedback.
DrJava: [http://drjava.org/](http://drjava.org/)
DrJava is a lightweight development environment for writing Java programs. It is designed primarily for students, providing an intuitive interface and the ability to interactively evaluate Java code.

JACard Flex: [http://www.jmu.edu/cardctr/flex.shtml](http://www.jmu.edu/cardctr/flex.shtml)
Put some ex money on your JACard to use for printing, which costs about $0.05/page. There may be times when you need to print from the lab.

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 10 hours per week outside of class or more).
- 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.

I will help you do this by:

- building tutorials and assignments to help you to learn new content. and
- creating interesting problem solving activities for class days, and
- creating challenging lab problems for your lab days, and
- creating larger problems that let you integrate a number of new skills, and
- supporting you as you learn this new material and tackle challenging problems

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. More information: [http://www.jmu.edu/syllabus/#Disability](http://www.jmu.edu/syllabus/#Disability)
**Attendance:** Since this class is based on POGIL activities and the process of engaging with other students and the material in class, attendance is mandatory. I will take attendance each day through a check off sheet (and turning in activity materials). 4 attendance days will act as one homework/lab/other assignment. This will be in addition to any material that may be collected from that day.

"**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. Students with an excused absence will also be exempted from the attendance policy above for that day.

**School closing for weather or other unexpected circumstances:** Watch our Canvas 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.

**Dates and deadlines:** [http://www.jmu.edu/syllabus/#AddingDropping](http://www.jmu.edu/syllabus/#AddingDropping) You are responsible for knowing and abiding by the registration dates and deadlines in effect for this and all other classes. See the link to those deadlines here.

**NOTE:** This syllabus represents a general plan for the course and deviations from this plan may be necessary during the duration of the course.