

CSCI 531: JAVA PROGRAMMING, Spring 2015

INSTRUCTOR:

Dr. Ray Maleh
Adjunct Professor, Department of Computer Science
Office: TBD, Phone: 903-408-2605
Email: Ray.Maleh@L-3com.com AND Ray.Maleh@tamuc.edu
Office Hours: Mondays 7:10 PM - 8:10 PM, Other Times By Appointment

CLASS MEETING TIMES:

Mondays 4:30 PM - 7:10 PM, EDS135
Jan. 20, 2014 - May 15, 2014

TEXTBOOK:

Horstman, Cay. Big Java 5th Edition. New York: John Wiley & Sons, Inc. (4th Edition is also acceptable)

CREDITS: 3

PREREQUISITES:

Some familiarity with programming; knowledge of C and/or C++ is especially helpful.

COURSE DESCRIPTION:

As the Java language was designed to facilitate object oriented programming, we begin with a thorough coverage of classes/objects, methods, console and file input/output, exceptions, and the concepts of inheritance, encapsulation, and polymorphism. We spend a significant amount of time covering the Java class hierarchy and discussing how to exploit the use of super-classes and interfaces to achieve coding efficiency, flexibility, maintainability, and generality. We then spend several weeks learning about the development of graphical user interfaces (GUIs) in Java, including the use of layout managers and design/installation of event listener objects. In the final weeks, we cover advanced topics including input/output streams, multi-threading/synchronization, internet networking, and database connectivity. At the conclusion of the course, the student will be expected to complete a project involving the design of a fairly complex Java program that consists of a GUI and utilizes at least two of the advanced programming areas.

STUDENT LEARNING OUTCOMES:

- (SLO531.1) Students will design and implement programs in the Java programming language that make strong use of classes and objects.
- (SLO531.2) Students will learn to print formatted text to the console output and read/parse console input text using a Scanner object.
- (SLO531.3) Students will apply logical constructs for branching and loops as well as use iterator objects when appropriate.
- (SLO531.4) Students will learn to define classes and methods. In addition, students

will learn the basics of polymorphism through use of super-classes and interfaces. Finally, students will develop an understanding of the Java language class hierarchy including the cosmic Object superclass.

- (SLO531.5) Students will learn to create and access arrays and array lists, including those with references to generalized objects types.
- (SLO531.6) Students will develop linked data structures such as lists and trees.
- (SLO531.7) Students will learn how to handle exceptions and errors. Students will design and implement custom checked and unchecked exception types.
- (SLO531.8) Students will become familiar with the use of input, output, and object stream objects. Students will use such streams for file processing as well as client/server communications tasks.
- (SLO531.9) Students will develop sophisticated, interactive user interfaces using the Java Swing class and appropriate layout managers.

Students will also be exposed to advanced topics including multithreading, internet networking, and JDBC database connectivity (time permitting).

METHOD OF EVALUATION (Tentative):

Your grade in the course will be calculated as follows:

Homework Assignments:	25 %
Quizzes:	25 %
Midterm and Final Exam:	25 %
Final Project:	25 %

Assignments and projects will be posted in the university's e-College communication system. Assignments must be turned in online in the appropriate drop box. Quizzes will be random and unannounced.

All code must be readable. What this means is that you indent nested statements and provide a generous amount of comments. As an example, consider the following two pieces of code, both of which calculate the factorial of an integer n:

```
//Good Code
//This function calculates the factorial of its non-negative integer input n. Output type is integer.
public static int factorial(int n)
{
    if (n == 0) //Base Case: 0! = 1
        return 1;
    else
        return n*factorial(n-1); //For n > 0, recursively let n! = n*factorial(n-1)
}

//Bad Code
public static int factorial(int n) {return n>0?n*factorial(n-1):1;}
```

While both are correct, the second version can be very difficult for another person to understand. Conversely, don't over-comment your code to the point where I'm struggling to find the locations of the actual Java statements. When

evaluating your code, I will take coding style into account. I believe that it is imperative for programmers to produce readable code, especially when working on massive team-based software development projects.

Your final letter grade will be determined as follows:

- A - total number of points ≥ 89.5
- B - $79.5 \leq$ total number of points < 89.5
- C - $69.5 \leq$ total number of points < 79.5
- D - $59.5 \leq$ total number of points < 69.5
- F - total number of points < 59.5

I reserve the right to curve the grades in the course; however, for a given raw average, you will at least earn the grade letter shown above (if not better). As you will notice, I have already incorporated a standard rounding scheme into the schedule of grades. Thus, please do not ask me to round your grade at the end of the semester.

At the end of every semester, there is always at least one student who asks to have his/her final grade changed due to some external heart-breaking circumstance (e.g. I need an "A" because a "B" will keep my core GPA below 3.0 and I can't graduate.) Don't try this on me! It is a waste of my time and your time. The only time I ever change a grade at the end of the semester is if there is an error on my part in grading.

ACADEMIC ETHICS:

"All students enrolled at the University shall follow the tenets of common decency and acceptable behavior conducive to a positive learning environment." (See Student's Guide Handbook, Policies and Procedures, Conduct).

ATTENDANCE POLICY:

Students are expected to be present at all class lectures. If a student is absent from class on the due date of any assignment, they are expected to make alternative arrangements to assure that the assignment is turned in ON TIME. Any student wishing to withdraw from the course must do so officially as outlined in the class schedule. THE INSTRUCTOR CANNOT DROP OR WITHDRAW ANY STUDENT.

COURSE REQUIREMENT DEADLINES:

Credit will be given for ONLY those exam(s), program(s), and/or project(s) turned in no later than the deadline(s) as announced by the instructor of this class unless prior arrangements have been made with the instructor.

ACADEMIC ETHICS AND HONESTY STATEMENT:

Scholastic dishonesty is a violation of the Code of Student Conduct. Scholastic dishonesty includes, but is not limited to, cheating on a test, plagiarism, and collusion. "All students enrolled at the University shall follow the tenets of common decency and acceptable behavior conducive to a positive learning environment." (See Student's Guide Handbook, Policies and Procedures, Conduct).

Academic dishonesty includes, but is not limited to, cheating on tests, plagiarism and collusion. **Cheating** includes copying from another student's test or homework assignments or projects or quizzes, using materials not authorized, collaborating with or seeking aid from another student during a test, knowingly using, buying, selling, stealing, or soliciting the contents of an unadministered test, and substituting for another person to take a test. **Plagiarism** is the appropriating, buying, receiving as a gift, or obtaining by any means another's work and the unacknowledged submission or incorporation of it in one's own written work. **Collusion** is the unauthorized collaboration with another person in preparing written work for the fulfillment of course requirements. Academic dishonesty is a serious offense in college. You will be given not only a failing grade on the assignment or test, but also a failing grade for the class. Further, it will result in suspension from college.

PLAGIARISM:

In any written paper or test or assignment or quiz or project including code and/or documentation, you are guilty of the academic offense known as plagiarism if you half-copy or copy another author's sentences, words or any part of the content. **This will result in an automatic grade of "F" for the course.** Hence any of these must be fully avoided in order not to fail the class. Students copying from work done in previous semesters by former students as well as copying from internet sources without proper referencing will result in you failing this course. You cannot mix the author's words with your own or "plug" your synonyms into the author's sentence structure. To prevent unintentional borrowing, resist the temptation to look at the source as you write. The author's words, phrases, sentences must be put in your words and in your way of writing! When you do this, you are demonstrating your ability to understand and comprehend the material!

STUDENTS WITH DISABILITIES:

Students requesting accommodations for disabilities must go through the Academic Support Committee. For more information, please contact the Director of Disability Resources & Services, Halladay Student Services Bldg., Room 303D, (903) 886-5835