INFSYS 3816/6808 (combined section) – Summer 2013

Managerial Applications Of Object Oriented Programming II

Monday/Wednesday 6:00 p.m. to 8:40 p.m. SSB 102

Instructor: Shaji Khan, Ph.D., MS-Computer Science, BA, B.Com.
Office location: 231 Express Scripts Hall (ESH)
Office hours: Monday/Wednesday 4:45 pm to 5:45 pm and other times by appointment
E-mail: shajikhan@umsl.edu **Cell phone:** (314) 489-9733 (no texts please)

Note: If anyone has a health condition or disability which may need my attention, please contact me and the Disability Access Services Office at 144 Millennium Student Center (ph: 314-516-6554).

Course Materials

1. There is no required text-book for this course
2. Optional but highly recommended Text:
    - Introduction to Java Programming, 9/E, by Y. Daniel Liang (either comprehensive or brief version)
    - (8th and 7th editions should also work just fine)
3. Free online books for additional reference:
    - Please see "MyGateway/Syllabus and Online Texts" for an updated list of free online text-books
4. Official Java Tutorial is the best source of updated information
    - Learning the Java Language: http://docs.oracle.com/javase/tutorial/java/index.html
    - GUI development using Swing: http://docs.oracle.com/javase/tutorial/uiswing/index.html
5. Other material as assigned in class or uploaded to MyGateway

Course Description

This is a combined section for the advanced java / internet programming courses at the undergraduate and graduate levels. This is NOT an introductory course in object-oriented programming. This course builds upon foundations of Object-oriented programming using Java and goes in depth covering advanced concepts. Topics covered will include composition, inheritance, polymorphism, abstract classes, interfaces, generics, and collections. In addition, this course will introduce one or more types of application development such as GUI/Desktop Applications or Web-based applications using Java Server Faces (JSF). It is assumed that students will be familiar with basic programming concepts using an object-oriented approach (preferably using Java). Grades will be assigned based on performance in programming assignments and examinations.

Grades

Letter grades will not be assigned to individual components of the course. Only points (numeric scores) will be assigned. These scores will be added at the end of the course to determine your Final Score out of 100. This "final score", **rounded to one decimal place**, will determine your overall letter grade per the schema listed below.

Grade Structure:

Attendance	10%
Exam 1	25%
Exam 2	25%
Assignments	40%
Max Possible Final Score:	100%

Grading Policy:

Final score ranges and corresponding letter grades				
>=94	**A**		**70-73.9**	**C**
90-93.9	**A-**		**62-69.9**	**C-**
86-89.9	**B+**		**58-61.9**	**D+**
82-85.9	**B**		**50-57.9**	**D**
78-81.9	**B-**		**46-49.9**	**D-**
74-77.9	**C+**		**<46**	**F (fail)**

The schema presented above will be applied strictly. For example, final score of 93.9 (rounded to one decimal place) is an A- and not an A. I will not entertain any arguments in this regard.

Attendance (10% of overall grade)
It is very important students attend all class periods. I strongly recommend it. It will be very difficult to keep up if you miss class. I will take attendance by passing a sign-in sheet. Please make sure you sign the attendance sheet for all days you are present in class.

I will allow two excused absences. No need to inform me. Beyond this, I will deduct proportional points from your attendance score for each additional day missed. A poor attendance score can dramatically affect your overall class grade. Your attendance score will be calculated as follows:

((# of Days Present / (Total # of Class Periods – 2)) * 100) **OR** 100, whichever is less.

[Where "# of Days Present" = the number of days you signed the attendance sheet]

Further, try not to be late for class. However, I encourage you to walk in even if you are late.

If I have to cancel class under extenuating circumstances I will do my best to inform you at least 1 day in advance via your UMSL email. Please check your UMSL email account regularly.

Homework Assignments (40% of your overall grade)
During this semester, I will assign between 3 and 6 assignments. These will be assigned when I feel you may benefit from practicing what was covered in class. I will upload detailed assignment documents to MyGateway and make relevant announcements in class. Programming requires active participation in class and plenty of practice. The assignments are designed to help you master basic programming concepts.

- Assignments will be posted under "Docs & Assignments/Assignments" within MyGateway.
- You will receive an "announcement" on MyGateway and an email notification (to UMSL email)
- Due dates will be listed in assignment documents. Late submission penalty is 10% for each 24 hour delay, until no points remain.
- Completed assignments will be submitted through MyGateway itself (I will provide instructions)

Exams (50% of your overall grade)
There will be a total of two (2) exams. The exams will assess your understanding of the concepts covered in class. The first exam will be about mid-way into the semester and the second exam will be at the end. See schedule (below) for dates.

- Exams will mostly entail students developing simple applications (i.e. writing code).
- Exams *may* include a conceptual component (mostly multiple-choice questions)
- **Exams will be take-home and "open-book"**.
- Exams will be comprehensive. If a topic/concept is listed on MyGateway course documents, then it is included for the exams.

Class participation
Participation in class from your end is crucial for you and your class mates. Unless under extenuating circumstances, please read assigned material **before** class and make every effort to participate in class discussions. My belief is that one cannot learn programming by just listening (however actively) to lectures or reading text-books. Programming is about actually writing code and learning by doing and making mistakes. Don't let a few students drive the conversations. Make sure you understand before we move on. "I will figure it

out later…" kind of attitude just won't work. I will make every attempt to allow class time for actually writing code. I will also be available for any help students need regarding assignments or class topics.

Code of Conduct and Academic Integrity: All work assigned during this course is to be carried out individually. Please refrain from copying code from others or from the Internet. I will deal with instances of academic dishonesty rather strictly (please see guidelines below). If I feel a student has copied code from another student, both the student providing the code and the student copying will receive no credit on the assignment(s) / exams. If I feel a student has obtained outside help then the student will receive no credit for the respective assignment(s) / exams. The onus lies on you to prove otherwise by undergoing an oral examination and/or completing code in a supervised setting. Again, I am available for help as many times as needed.

Academic Dishonesty Guidelines: (from Academic Affairs website, revised May 2003)
Students at the University of Missouri-St. Louis are expected to exhibit the highest standards of academic integrity. An act of academic dishonesty is an offense against the university. For that reason, university rules prescribe disciplinary as well as academic consequences for academic dishonesty. The Board of Curators recognizes that academic honesty is essential for the intellectual life of the University. Faculty members have a special obligation to expect high standards of academic honesty in all student work. Students have a special obligation to adhere to such standards. In all cases of academic dishonesty, the instructor shall make an academic judgment about the student's grade on that work and in that course. The instructor shall report the alleged academic dishonesty to the Primary Administrative Officer.

The bulletin and student planner provide guidelines for appropriate academic conduct. The following procedures guide the campus in upholding the University of Missouri Collected Rules for Student Academic Conduct. Academic Dishonesty is defined by Collected Rule 200.010 STANDARD OF CONDUCT
http://www.system.missouri.edu/uminfo/rules/programs/200010.htm

NOTE: I will make announcements on MyGateway and I strongly encourage you to visit this course under MyGateway regularly for important updates and documents.

Please also check your UMSL email account regularly.

Tentative Schedule

Required readings for each week will be posted under "Docs & Assignments" of this course on MyGateway. Please prepare fully for each class by finishing the assigned reading prior to class.

Class	Date	Topic
1	May 20	Course Introduction, Syllabus, Course Preparation, Review of Object-oriented approach
2	May 22	Review of Object-oriented basics, Classes, Objects, Constructors, Encapsulation, Types of Members Overview of Unified Modeling Language (UML)
3	*May 27*	*No class, Memorial Day holiday*
4	May 29	Object-oriented Thinking: some guidelines, overview of UML (if needed)
5	June 3	Developing classes that model composition relationships Introduction to inheritance
6	June 5	Inheritance contd. Polymorphism
7	June 10	Exception Handling and Text Input / Output (I/O)
8	June 12	Introduction to Collections (with a brief overview of Generics) **Take-Home Exam 1**: *Available* on MyGateway starting June 12 11:59pm Central Time
9	June 17	(no class meeting)
		Take-Home Exam 1: *Due* (submission deadline June 19, 11:59pm Central Time)
10	June 19	Abstract Classes Interfaces (NOTE: Instructor will not have office hours on this day)
11	June 24	Revisit Collections (this time with knowledge of interfaces) Difference between Abstract Classes and Interfaces (when to use each) Graphical User Interface (GUI) Development Basics using ***Swing***
12	June 26	More GUI Basics GUI: Event driven applications GUI: Some useful GUI components
13	July 1	GUI: Additional GUI components
14	July 3	Overview of Java Persistence API, Database Programming (Tentative)
15	July 8	Building a Simple Web Application, Java EE, EJB, JSF (Tentative) **Take-Home Exam 2**: *Available* on MyGateway starting July 8 11:59pm Central Time
16	July 10	(no class meeting)
		Take-Home Exam 2: *Due* (submission deadline **July 14**, 11:59pm Central Time)

NOTE: There will be no make-up exams. Please follow the exam due dates carefully.