**Course Syllabus**

|  |  |
| --- | --- |
| **Instructor:** | James Papademas, MBA, MSMC, MISM |
| **Cell Phone** | 773.558.7866 |
| **Email:** | IIT E-Mail: [jpapadem@iit.edu](mailto:jpapadem@iit.edu) |
| **Office:** | Rm. 223 PH at IIT Chicago Campus /per pre-arranged appointment. |
| **Office Hours:** | M - 12:45-1:45, W - 12:45-2:45 |

Required text:

Starting Out with Java: From Control Structures to Data Structures, Tony Gaddis, 4th

Online docs:

Java Learning Trail - <http://docs.oracle.com/javase/tutorial/java/>

Beginners Reading

If you have little experience programming, or need to catch up on some basic concepts.

Think Java - <http://www.greenteapress.com/thinkapjava/html/index.html>

Resources

Java 8 API - <http://docs.oracle.com/javase/8/docs/api/>

How to search the Java API for String Class – <http://docs.oracle.com/javase/8/docs/api/java/lang/String.html>

Javaranch is a great place for beginners to ask questions

<http://www.javaranch.com/index.jsp>

Advice to students from Top Developers

<http://www.oracle.com/technetwork/articles/javase/index-136000.html>

mkyong is a good place for working Java examples (unit testing example)

<http://www.mkyong.com/unittest/junit-4-tutorial-1-basic-usage/>

Advanced Reading

Effective Java, Josh Bloch

Starting Out with Java: From Control Structures to Data Structures, Tony Gaddis, 3rd [recommended]

Clean Code: A Handbook of Agile Software Craftsmanship

<http://www.amazon.com/Clean-Code-Handbook-Software-Craftsmanship/dp/0132350882/ref=sr_1_1?s=books&ie=UTF8&qid=1375411596&sr=1-1&keywords=clean+code>

Inspirational reading

Java Developer Salary Info for Chicago

<http://www.indeed.com/salary/q-Java-Developer-l-Chicago,-IL.html>

How to become a hacker

<http://www.catb.org/~esr/faqs/hacker-howto.html#why_this>

Teach Yourself Programming in 10 years

<http://www.norvig.com/21-days.html>

Java Video Tutorials

<http://www.caveofprogramming.com/>

**Prerequisites: [(ITM 311 with min. grade of D)]**

**Course Description**

This course covers object oriented programming concepts in the Java Standard Edition

platform. Employing the latest software development kit, the student considers software

development topics in data structures, stream I/O, serialization, concurrency and graphical

clients. Software engineering topics including packaging, deployment, debugging and unit

testing. Hands on exercises reinforce concepts gained throughout the course. A final project

integrates course topics into a contemporary Graphical User Interface client application.

**Course Objectives**

Students completing this course will be able to:

● Write Object Oriented Java SE code.

● Create a Java based Graphical User Interface.

● Locate application functionality from a large programmer API.

● Author well constructed code and software documentation.

● Utilize an IDE to develop, test and debug Java SE code.

**Course Outcomes**

Students completing this course will be able to:

● Understand basic Object Oriented programming concepts.

● Apply Test Driven Development methodologies.

● Understand packaging and deployment Java SE applications.

● Describe Software development terminology.

**Course Activities**

You will meet the objectives listed above through a combination of the

following activities in this course:

● View and comprehend all screen recording course content.

● Complete all course assignments at a proficient level with high quality and on time.

**Course Requirements**

Student Responsibilities: Class attendance and active participation are essential if students are to receive maximum benefit from the class. Participation requires preparation including completion of reading, labs, projects and exams by the due dates. If you cannot attend class or complete assignments, labs, projects or exams on time, please let the instructor know beforehand so that we can discuss alternative strategies. It is the student’s benefit to use their time wisely whether it is in preparation for class, during scheduled class, or in the lab. When students are in any IIT lab environment, they should abide by the college policies. Questions and comments are welcome.

Exams and make-up policy: There will be a mid term and final exam for the course. No **retakes** of exams are allowed unless there are extraordinary circumstances. Any exams may be taken early if the instructor is given adequate time to prepare testing arrangements.

Assignments & General Grading: It is extremely *critical* that students complete all assignments timely otherwise late points will be deducted accordingly. Submitting assignments timely in the order assigned will ensure progression according to the academic design of the course. Handing in of **bulk assignments** will NOT be acceptable. The only way to learn Java is to code in Java. The best Java programmers are the ones who have invested the time to learn the concepts and applied them to programming problems. Please do not expect to finish the assignments for this class the night before, or during the weekend they are due. The project assignments will take considerable effort. “A” grades on projects are reserved for code that properly fulfills all of the listed requirements in a computationally accurate and reasonably efficient manner, and is well organized and readable based on the basic design principles covered in class. Also class participation is compulsory for that A grade as well.

Email: Every attempt will be made to answer email daily. Please indicate in your email clearly the problem you are experiencing in your subject and body of your email. Please also include your name and course enrolled.

Academic Policy: Any violations of IIT policies regarding academic honesty and or integrity will be referred automatically to the appropriate college authorities for disposition. Please see appropriate pages in the college catalog for definitions and regulations. The minimum penalty for cheating will be a zero for all parties involved on that exam, assignment, lab, project or quiz.

Withdraw policy: No longer attending a class does not constitute an automatic withdrawal. Students are expected to withdraw from the course if they have decided not to pursue the course anymore.

Classroom behavior: During the class time, *considerate* conduct by all persons is important to a favorable learning environment. Taking in class outside of interacting with classroom discussions is a disruption! Any infringement on the rights of others to get an education will be dealt with in an appropriate manner. Please set all electronic devices such as cell phones or pagers to silent or vibrate mode. No cell phone talking is permitted in the classroom. If you must take the call, please continue your conversation outside of the classroom and please make it short so as to not miss your lecture material. Point reductions can result from negative behavior and negative influences to others.

General notes: In order to achieve the course objectives, it is important to enjoy the class in addition to complying with the above requirements, and the rules and policies of IIT. Most students sign up for the courses with the best intentions. If you are experiencing course/college related problems, please feel free to discuss it with your instructor before a crisis develops so we can resolve them in a manner beneficial to all parties involved.

Reasonable accommodations will be made for students with documented disabilities. In order to receive accommodations, students must obtain a letter of accommodation from the Center for Disability Resources and make an appointment to speak with me as soon as possible. The Center for Disability Resources (CDR) is located in the Life Sciences building, in room 218, with telephone 312-567-5744 or with email at [disabilities@iit.edu](mailto:disabilities@iit.edu).

**Grading and Evaluation Criteria.** Grade distribution is represented as follows:

A – 90% and up

B – 80 to **89.99**%

C – 70 to **79.99**%

D – 60 to **69.99**%

F – 59.99% and below

**Grading Policy**

Visit the various assignment links in Blackboard for details about each assignment listed below.

Be sure to pay close attention to deadlines. There will be no make up assignments or quizzes, or late work accepted without a serious and compelling reason and instructor approval. Points you receive for graded activities will be posted to the Blackboard Grade Book. Click on the My Grades link on the left navigation to view your points. The instructor or TA will update the online grades each time a grading session has been complete—typically 10 days following the completion of an activity. You will see a visual

indication of new grades posted on your Blackboard home page under the link to this course.

Final grades assigned for this course will be based on the percentage of total points earned and

are assigned as outlined above including other soft skill participation. See [ITM Student Handbook](http://www.itm.iit.edu/resources/studentresources.php) for grade percentage to letter grade table.

The class has the following grade weight based on a tentative point scale

Midterm- 100 points

Final exam- 200 points

Final project- 200 points

Labs 400 points

Total points- **900** points

Incomplete Grading Policy

Under emergency/special circumstances, students may petition for an incomplete grade. An

incomplete will only be assigned based on department protocol including instructor approval. All

incomplete course assignments must be completed the following term as noted in the IIT Academic Calendar.

**Academic Dishonesty**

Academic Dishonesty is not acceptable and will not be tolerated in ITMD 411. Any assignment found to be

Plagiarized and found to be copied will result in a **zero** grade to all parties involved. Please read the ITM

Student Handbook to review the department's policies on plagiarism and identical or substantively identical work. NO "GROUP" WORK IS TO BE CONDONED OR SUBMITTED UNLESS ASKED FOR- THIS COLLABORATION IS CONSIDERED PLAGIARISTIC!

Grading for lab assignments will be based on the following criteria and point allocations:

**Program correctness**: 60 points (Your program runs and executes without errors, meeting all program requirements with readable program output display)

**Design Approach and Documentation**: 30 points Program must follow standard programming style. Please examine programming styles from class demo’s, textbooks, proper usage of blocks and indentations, proper documentation, meaningful variable names, comment statements, algorithm development, and programming logic used/approach to resolve assigned problem. **Label each lab with your name at the top of your source code as well as your lab number!!! (Each lab must have adequate snapshots of output for full credit as well.**)

**Program enrichments**: 10 points Error proof program, extra features included, OOP methodology consideration, reliability and ease of maintenance-above and beyond.

**Technology Requirements**

● A Windows, Mac or Linux computer, preferably with a Core 2 Duo or better processor,

and 2 or more Gigabytes of ram. The Eclipse v4.6 Integrated Development Environment

for Java and the Java Development Kit (both are free software).

● Internet Connection (DSL, LAN, or cable connection desirable)

● Access to IIT Online System (blackboard.iit.edu)

**Blackboard – The IIT Online Classroom**

We will use IIT's Blackboard system (<http://blackboard.iit.edu>) to communicate weekly agendas, submit homework, labs, ask questions, to post lecture materials and get feedback. Each student should have been notified of his or her Blackboard account for this course. If you have not been notified, go to above web page where there is contact information. Blackboard weeks start from Monday through Sunday.

**Tentative Schedule of Topical Coverages**

|  |  |
| --- | --- |
| **Week** | **Topic(s)** |
| 1 | Java SE Essentials, algorithm development, the Java compiler/Eclipse IDE |
| 2 | Object-Oriented Programming (OOP) concepts / OOAD (OOP Designs) |
| 3 | Arrays and Arraylists |
| 4 | Abstraction, Inheritance, Polymorphism |
| 5 | Stream I/O  Exception Handling |
| 6 | Generics  Collections |
| 7 | Collections  Stream API |
| 8 | Linked Lists  Stacks, Queues |
| 9 | **Phase I final lab project (Design Phase)**  Mid Term Review  **Mid-term** (in class, closed book/notes, devices, etc.) |
| 10 | Databases |
| 11 | GUI’s – SwingX  **Phase II final lab project (Logic Phase)** |
| 12 | Serialization  Networking |
| 13 | Regular Expressions  Packaging and Deployment |
| 14 | Concurrency  JUnit testing / TDD |
| 15 | **Final Exam Study** |