ITM 692: Special Topics in Information Technology
University at Albany, State University of New York
Fall 2006 Syllabus

INSTRUCTOR INFORMATION
Name: Sanjay Goel / Peter Duchessi
Email: goel@albany.edu / p.duchessi@albany.edu
Phone: (Goel) 442-4925 / (Duchessi) 442-4945
Office Location: (Goel) BA310b / (Duchessi) BA 312
Office Hours: (Goel) M: 12:30-2:00pm or by appointment / (Duchessi) TH: 12-2pm or by appointment

CLASS INFORMATION

Time:	TH 9:00am-12:00pm
Location:	BA 233
Dates:	(Java) September 7 – October 5 / (Special Topics) October 12 – December 7
Credit(s):	3
Call #:	6434
Available Lab(s):	MIS (to the right of BA 233) and HRIS (to the left of BA 234) Labs

COURSE WEBSITE:

Java: http://www.albany.edu/~goel/classes/fall2006/itm692/
Special Topics: http://webct.albany.edu:8900

These course websites should be your main source of course material for their respective parts of the class and contain all relevant course information including details on grading, projects, assignments, course schedule, etc. In addition, they should provide a "living syllabus" a will reflect any changes made to this document.

TEXT & REFERENCE BOOKS
Text: Keogh, Jim. (2004). Java DeMYSTiFieD. McGraw-Hill/Osborne. ISBN: 0-07-225454-8

COURSE OVERVIEW
This course provides an overview of some emerging techniques in Information Technology and teaches concepts of advanced programming languages. The content of the course will change from year to year as new technologies emerge. The class this year will cover three separate topics, that is, Java programming, and guest lectures on various information technology-related topics. The initial third of the class will be taught by Professor Goel and the last two-thirds of the class will be taught by Professor Duchessi. The first part of the class focuses on development of simple business logic in a structured form. The focus is on development of logic rather than the specifics of a programming language. The class covers the basic elements of a programming language, such as data types, loops, arrays, functions etc. In addition, the basic concepts of object-oriented programming, such as, abstraction, polymorphism, and inheritance are discussed. By the end of the course, students should be able to write simple programs in the JAVA language and be able to abstract a problem into a class structure.

LEARNING OBJECTIVES
Students will learn:
1. Critical thinking and logic skills for problem solving
2. Syntax of Java language
3. Some concepts of object oriented programming
4. Steps involved in writing software

5. Basic tools for writing, compiling, and running programs

Students should be able to:
1. Install the programming environment for programming in Java
2. Write algorithms for simple problems
3. Compile, debug, and run Java programs
4. Create classes in Java
5. Gain further understanding into advanced topics in IT.

ACADEMIC INTEGRITY

All students are expected to follow University at Albany guidelines on academic integrity (see the Academic Integrity section of the course site for more detail). Whenever you come to me with a special request, think about whether your request is unfair to the other students. I am willing to do anything to help as long as I feel it will be useful to you and I make sure that it is fair to all students in the class.

GRADING

Whenever you come to me with a special request, think about whether your request is unfair to the other students. I am willing to do anything to help as long as I am fair to all students in my classes. There will be no make-exams or other accommodations for lateness unless there is a valid medical excuse or other similar emergency.

Assignments (25%): Assignments given in any week are due at the beginning of the class next week. There will be a penalty of 10% per day for late assignments unless there is a very pressing reason for the delay. In-class and homework assignments should be done in groups of two (assigned at the start of class). Assignments are typically 5-10 points each and will consist of exercises relevant to the material discussed in class. Please see the Assignments section of the course site for further details and guidelines

Project (35%): Projects should be done in groups of two (not assigned) and will feature of JAVA programming project based on guidelines. For more details and guidelines, please see the Projects/Papers section of the course site.

Exam (40%): The exam will consist of multiple sections (essay-style and short answer) in which you will have to apply a majority of what has been learned during the semester in order to assess individual performance. A sample exam and solution set will be provided for review.

Teams (Pair Programming): This year, students will be introduced to the concept of pair programming where a pair of students works side-by-side collaborating on software development. At any given time, one student is the driver and has the control of the computer and is actively writing the code while the other student acts as an observer and partner who is continuously monitoring the work of the other student to identify syntactic errors and algorithmic correctness. The two students switch roles periodically so that both get the experience of program solving as well as programming. The teams would thus include two students each for both the project and assignments.

JAVA PHILOSOPHY

The goal of this class is to promote logical thinking in students while learning the syntax, semantics, and pragmatics of a programming language. The language chosen for this course is Java because of its versatility and acceptance in the software development community however any other object oriented language could be used for a similar learning experience. Even though most students in the MBA program are not expected to pursue a career in software development, it is important to learn computer programming. Learning programming not only provides you with the syntax and semantics to write

instructions for a computer but develops fundamental thinking and problem solving skills. While writing a computer program the problem needs to be broken down into intricate steps, which ensures clarity of thought for the code writer. The programming process requires defining, analyzing, developing an algorithm, writing the language syntax, and debugging the program. The process of defining the problem teaches a student to articulate the problem precisely, and the processes of analyzing the problem and developing the algorithm require the writer to examine several alternative solutions that hones critical thinking skills in the user. The step of writing the code requires translating the algorithm into the syntax of the code and is perhaps the simplest of the tasks. Most students assume that the root cause of their frustrations in programming stems from a lack of familiarity with the syntax however on the contrary the fundamental problem lies in a lack of ability to think clearly. The process of debugging requires tracing through the program and identifying the root cause of an error a skill that will be required time and again in the business world to identify causes of potential failures and for examining risks of failures.

PROGRAMMING ENVIRONMENT
To develop software in Java three things are required: 1) an *editor* to write programs, 2) a *JAVA Compiler*, and 3) an *execution environment*. There are several good editors that can be used including TextPad, WordPad, and Notepad (Windows based) and Emacs, which is Unix (or Cygwin) based system that runs on top of Windows. The advantage of TextPad is that it provides a menu bar that allows you to compile the code within the editor itself. The java compiler should be installed based on the software installation instructions that are provided to you. The compilation and execution should be done using a Command shell or Cygwin shell. Please refer to the accompanying instructions to ensure that your environment is correctly set. Some basic knowledge of Unix will be useful in the installation of the software and setting up the environment.

COURSE SCHEDULE

Date	Topics	Class Notes	Readings	Assignments
9/7	Java Development Environment, Programming Fundamentals, Data Types, Operators, Expressions, Simple IO	To Be Announced	Chapters 1-3	To Be Announced
9/14	Control Flow and Statements, Functional Abstraction (methods), Arrays		Chapters 4 & 5	
9/21	Data Abstraction: (Constructors, Scope of Variables & O-O Design)		Chapter 6-8	
9/28	Inheritance, Polymorphism		Chapter 9 & 11	
10/5	Exam I		Chapters 1-9, 11	
10/12	Special Topics: To Be Announced	Provided In-Class	To Be Announced via WebCT	To Be Announced via WebCT
10/19				
10/26				
11/2				
11/9				
11/16				
11/30				
12/7				