The Proposed Pitney Bowes ECE MS Program

by

Douglas Lyon and Jerry Sergent,
Co-Directors of the ECE MS Program
{lyon@docjava.com},{sergent@mail.fairfield.edu}

The following is a proposed program to be offered on-site at Pitney Bowes by Fairfield University's School of Engineering. This program leads to a Master of Science in Electrical and Computer Engineering.

1. Characteristics of ECE Graduates

Electrical and Computer engineering embodies the science and technology of design, construction, implementation, and maintenance of software and hardware components of modern electrical, electronics and computing systems. The discipline of has emerged from the traditional fields of electrical engineering and computer science as a separate, although intimately related, discipline.

Our graduates are solidly grounded in the theories and principles of computing, mathematics, science, and engineering and apply these theories and principles to design hardware, software, and processes and to solve technical problems. Continuing advances in computing and electrical systems have created opportunities to apply these developments to a broad range of applications in engineering.

Our graduates work in most industries, including the computer, aerospace, telecommunications, power, manufacturing, defense, and electronics industries. They design high-tech devices ranging from tiny microelectronic integrated-circuit chips, to powerful systems that utilize those chips, to efficient telecommunication systems that interconnect those systems. Applications include consumer electronics (CD players, televisions, VCRs, stereos, microwaves, gaming devices), advanced microprocessors, peripheral equipment, and systems for portable, desktop and client/server computing, communications devices (cellular phones, pagers, personal digital assistants), distributed computing environments (local and wide area networks, wireless networks, Internets, Intranets), embedded computer systems (such as aircraft, spacecraft, and automobile control systems in which computers are embedded to perform various functions), and a wide array of complex technological systems, such as power generation and distribution systems and modern computer-controlled processing and manufacturing plants.

In addition to mathematics and science, our graduates may have a solid foundation in electronics, logic design, computer organization and architecture, and networking, as well as an understanding of software design, data structures, algorithms, and operating systems.

Our students get a strong laboratory experience, using modern laboratory environments,
industry standard hardware and software design tools. The ECE program fosters creativity, developing strong problem solving, design, and analysis skills with the ability to think logically.
Our students have good verbal and written communication skills.
Finally, our graduates realize that technology is rapidly changing, and therefore, to be successful, they must to continue to learn throughout their careers.

2. SW408 Java for Programmers I

School of Engineering Graduate Course Catalog 2000-2001:
A programming course introducing the fundamentals of Java to experienced programmers. Coverage includes the Java elements: objects, classes, variables, methods, syntax, reserved words, data types, operators, control structures, and container data structures. Object-oriented programming is integral to the course and is taught throughout. Accordingly, the concepts of encapsulation, inheritance, polymorphism, packages, interfaces, and inner classes are presented. The course teaches screen design using classes and graphics from Sun's Application Programming Interface (API). Data handling taught includes input from the keyboard, output to the screen, input from files, and output to files. Finally, the concept of multi-threading is introduced in preparation for follow-on studies. Lab included. Prerequisite: Significant programming experience or CS132 or CS134). Note: This course may be substituted for SW403 in order to meet the requirements for core courses.
Note: This course may be substituted for SW 403 in order to meet the requirements for core courses.

Textbook: *Java for Programmers, by D. Lyon*.
Computer Usage: Students **MUST** have access to a computer with a Java compiler. E-mail access is required.
Course Notes: Handouts/diskettes/e-mail
When: Fall Semester, Monday, 6:30- 9:20, Thursday, 6:30-9:20, Saturday 9AM – 12PM
Where: Bannow Science Center, Room 257
Who: Mr. Marquis (Saturday) Dr. D. Lyon (Monday and Thursday)

Phone: (203)641-6293
Fax: (203)877-4187
E- mail: maynard_marquis_99@yahoo.com, lyon@docjava.com
Web Page: http://www.docjava.com
Office Hours:
Monday, Wednesday or Thursday; 6 pm - 6:30 pm in class room (Dr. Lyon)
Monday, Wednesday or Thursday; 5 pm - 6:00 pm in Mc108 (Dr. Lyon)
Or by appointment (Dr. Lyon).
Saturday, 8:30 – 9:00 AM in classroom (Mr. Marquis)

Course Learning Goals: 1 – To teach how to write computer code in Java.
2 – To provide the means to leverage object-oriented programming.
3 - To prepare the student for advanced courses in Java.

Student Activities: Learning a new computer language is very much a hands-on activity, which cannot be learned from lectures or textbook reading alone. It does require those lectures and textbooks, but the real learning results from the laboratory trials and the homework assignments. To achieve the course objectives, the student must have good

class attendance and participation, conduct the computer programming tasks during the laboratory periods as well as the assigned homework. Homework assignments and laboratory trials are due at the beginning of the class following the assignments. They are to be placed in an envelope containing the student's name. The contents of the envelope will be a diskette and a paper copy of the requested Java source code.

Course Requirements: The schedule of activities and topics to be covered each week are outlined below. Each week will begin with responses to questions and a brief review on the previous week's topics. The first week will begin with administrative announcements and a review of this syllabus.
Join the List: Every student should join the e-mail list for the course, available on the bottom of the Visual Programming in Java SW408 web page at:
<http://www.docjava.com/java/sw406/syllabus/jointhej.htm>

Week 1- Introduction to Java and Object-Oriented Programming
1 - Java Introduction including the Java & HTML Models, What is Java?
2 - O-O Design including, Objects, Classes, Class Concepts and Class Hierarchy.
3 - Java Syntax and syntax example
4 - Java Packages, Application structure, Class structure, and Method structure.

Goals: To define the characteristics of O-O design and show an O-O example the first day. To provide an understanding of class hierarchy. To introduce those portions of Java Basics necessary to show the O-O example.
Outcome: Students can write and run hello world.
Outcome: Students can set up a small class hierarchy, with mammals, humans, students and professors.
Outcome: Students can display a class hierarchy graphically using the code warrior IDE.

Week 2 - Java Data Types and JavaDoc
1 - Reserved Words, Case Sensitivity
2 – Reference vs. Primitive Data Types
a – signed, vs. unsigned
b – fixed, vs, floating
c – 8 primitive data types
3 – Operators & Control Structures
4 – Introduction to JavaDoc
a – Introduction
b – JavaDoc tags
c – JavaDoc in CodeWarrior
d - Summary

Goals: To introduce basic elements of the Java language, which permit the writing and documentation of simple programs.
Outcome: The student will be able to write a Java program that demonstrates a class hierarchy, inheritance, and a simple degree of polymorphism. He or she will also be capable of documenting the program code using JavaDoc.

Weeks 3 – Java Basics
1 – Classes, Overloaded Methods, Constructors

2 – Getter and Setter Methods, Casting
3 – Reserved Words, null and super
4 – Modifier Static (methods & variables)
5 – Modifier Abstract (Classes & methods)

Week 4 - Important Java Concepts

1 – Reserved Word, this
2 – Interfaces (implementing in classes and extending other interfaces, more inheritance & polymorphism).
3 – Summary

Goals: To teach the student the concepts of static and abstract objects, and of interfaces, important tools in Java for Object-Oriented programming.
Outcome: The student will be able to write a program that demonstrates the use of static variables and interfaces in object-oriented programming, using a customer billing problem.

Weeks 5 - Important Java Concepts

1. packages
2. Modifier Visibility, Class Scope
3. Inner Classes (and again class scope)

Goals: To provide the student with the Java concepts that give object-oriented programming its modularity and extendibility benefits.
Outcome: The student will be able to control the scope of a variable or method. He will be able to use the concept of Abstract Methods and Interfaces to amplify and further demonstrate O-O inheritance and polymorphism.

Week 6 - Wrapper Classes Arrays and Vectors

1 – Methods in their Wrapper Classes, pp. 100-104,
2 – Casting, pp. 275
3 – Arrays – their construction, passing variables, pp. 407-67
4 – Vector Class and its methods, pp. 446-447

Goals: To provide the student with the means to manipulate data within his or her program, converting from one data type to another. To also provide the student with a means of grouping the data into the containers, arrays and vectors.
Outcome: The student will be able to write Java programs containing large amounts of data in an efficient manner. He will also be able to increase and demonstrate the efficiency, in terms of lines-of-code, of O-O programming using the containers.

Week 7 - Strings and Processing

1 – String Class
2 – String constructors
3 – String methods
4 – StringBufferClass

5 – StringTokenizerClass

Goals: To provide the student with the means to work with text-type data in Java programs.
Outcome: The student will be able to create, read, and parse text using Java programs.

Week 8 - Review for the midterm

Week 9 – Midterm Exam. An objective test, with multiple choice, fill-in the blank and code sample type questions.

Week 10 - Review Midterm Exam, Exceptions

1 – The try block
2 – throw and throws
3 – The catch block
4 – The finally block

Goals: To expose and explain Java exception handling to the students.
Outcome: The student will be able to create exception handling in Java programs, and he or she will be able to respond to requests for exception handling when using someone else's code.

Week 11-12 – Graphical User Interface Components and Listeners

1 – GUI Components: Frames, Labels, Buttons
2 – Event Listeners: WindowListener, ActionListener

Goals: To give the student the components necessary to create computer screen displays and to respond to events occurring on the screen.
Outcome: The student will be able to place GUI components onto a computer screen and to respond to events such as a mouse click or the push of a button.

Week 12

– Container Layouts: FlowLayout, GridLayout

Goals: To provide the means of arranging and manipulating GUI components in desired patterns on the computer screen.
Outcome: The student will be able to design functional and aesthetic displays on the computer screen using Java classes and using CodeWarrior.

Week 13 - File I\O, input & output streams

1 – Class hierarchy of java.io
2 – Input Stream, OutputStream
3 – Reader, Writer

Goals: To teach the student the techniques used in Java to transfer data to and from computer files.

Outcome: The student will be able to create files and read from computer files using Java. In particular, he or she will be able to create or read from files in the ASCI text format.

Week 14 - review for the final

Week 15 – Final Exam, An objective test, with multiple choice, fill-in the blank and code sample type questions.

Morelli Chapter 12, Recursive Problem Solving and Chapter 16, Data Structures, are not peculiar to Java and will only be covered as time permits. Some of the Data Structures, Arrays and Vectors, are covered in Week 6. Morelli Chapter 4, Applets, is not essential in a fundamental course and will not be covered. Chapter 15, Sockets and Networking, will not be covered in this course, and is relegated to an advanced course.

Additional References:

1 – Sun's Application Programming Interface, available in CodeWarrior.
2 – *Java How to Program*, Deitel & Deitel, Prentice Hall, Third Edition, 1999.
3 – *The Java Class Libraries*, Chan and Lee, Addison Wesley (c) 1997.
4 – *Java Source Book*, Ed Anuff, The John Wiley and Sons, Inc., First Edition, 1996.
5 – *The Java Programming Language*, K. Arnold & J. Gosling, Addison Wesley, Second Edition.

Grading:
1/3 Homework and Projects
1/3 Midterm
1/3 Final

Computer Usage:
Web and e-mail access are ***REQUIRED.*** You **MUST** have access to a computer with a Java compiler.

References:
The Java Class Libraries by Chan and Lee, Addison Wesley (c) 1997. See http://www.docjava.com for ordering this and other books.
Java Source Book by Ed Anuff, The John Wiley and Sons, Inc., First Edition, (c)1996.
Course Notes, as required, available on-line at http://www.docjava.com or by purchase of copies.
The Java Programming Language, Second Edition, by Ken Arnold and James Gosling, Publisher: Addison-Wesley, ISBN: 0-201-31006-6
Java 1.1 Unleashed, by Morrison, ISBN 1575213613
Core Java 2 Volume 1:Fundamentals by Cay Horstmann,Gary Cornell, ISBN: 0130819336
Java 1.2 Developer's Handbook by Philip Heller Simon Roberts, ISBN: 0782121799
Digital Signal Processing Recipes in Java by D. Lyon and H. Rao, M&T Books, (c) 1998.
Image Processing in Jav *a* by D. Lyon (c) 1999.

Goals:

To learn how to design large-scale software systems using Java. Object orientation and packaging are stressed. Testing and coding of several projects, with a variety of modern techniques.
Coordinator:
Douglas Lyon, Professor of Computer and Software Engineering. *E-mail*: lyon@docjava.com

Assignments will be due at the beginning of class. Assignments handed in after the beginning of class will loose 5 points. Assignments handed in after the end of class will lose 10 points. Late homeworks lose 10 points per day late, weeks ends and holidays included. Missed tests result in a zero unless a written excuse is presented.
More details are available about the class at: http://www.docjava.com

Homework requirements:
Print out a listing of the program. Print out the program input and output. You may need to do this at various levels of detail. Hand in a labeled disk with a printout. Place the disk in a #10 letter envelope and staple the envelope to the printout.
In the event that you *must* miss class, you may hand the homework in via a friend or another student. Should you find that you cannot find anyone in time, you must fax the homework in. Attachments will not be accepted. The fax number for homework is:(203)254-4013 fax. This is the School of Engineering fax machine, so please bring the homework to my attention so that it gets routed to me.

3. SW409 Java For Programmers II

Description
A second Java programming course covering more advanced Java programming. Topics include: Using Javadoc, Exceptions, Threads, Packages, nested and inner classes with an emphasis on sound object-oriented design using design patterns, Introspection, I/O, Persistence and advanced API topics, as time permits.

Prereq: SW408, or permission of the instructor.
Textbook: *Java for Programmers, by D. Lyon*. Copies are available from the SOE office**Computer Usage:** Students **MUST** have access to a computer with a Java compiler. E-mail access is required. **E-mail** access is required.
Course Notes: Handouts/diskettes/e-mail, web page
When: Wednesday, 6:30-9:15 pm,
Where: Mc102
Who: Prof. Lyon
Phone (203)641-6293
Fax (203)254-4013 fax
Web: http://www.DocJava.com
Office Hours: **Tuesday, Wed. Thurs; 3:30 pm - 5:30 pm in Mc209 (Dr. Lyon)**
Or by appointment (Dr. Lyon).

Pre-req by Topic

15. Threads
15.1 Concurrency, Multi-tasking, and Multi-Threading,
15.2 Making a new Thread
15.3 Thread States
15.4 Synchronized and Too Much Beer
15.5 Thread Groups
15.6 The Thread Manager
15.7 ThreadUtil
15.8 The Thread Queue
15.9 Running Thousands of threads
15.10 The Poor Mans' Thread
15.11 The Job Thread
15.12 Summary
15.13 Exercises
16. Nested Classes and Interfaces
16.1 The Member Inner Class
16.2 The Local Inner Class
16.3 The Anonymous Inner Class
16.4 The Nested Static Class
16.5 The Nested Interface
16.6 Summary
16.7 Exercises (to be added)
17. Readers and Files
17.1 ReaderUtils
17.2 The Dialog Class
17.2.1. Class Summary
17.2.2. Class Usage
17.3 The FileDialog Class
17.3.1. Class Summary
17.3.2. Class Usage
17.4 Futil Helper Methods
17.4.1. Futil.getReadFile
17.4.2. Futil.getWriteFile
17.4.3. Using Swing to get a File
17.4.4. Futil.getDirFile
17.5 The File Class
17.5.1. Class Summary
17.5.2. Class Usage
17.6 The FilenameFilter interface
17.6.1. Class Summary
17.6.2. Class Usage
17.6.3. DirFilter
17.6.4. The FileFilter Class
17.6.5. The WildFilter Class
17.7 The Ls Class
17.7.1. Ls.getWildNames
17.7.2. Ls.wildToConsole
17.7.3. Ls.getDirName
17.7.4. Ls.deleteWildFile
17.7.5. Ls.WordPrintMerge
17.8 Dir Lister
17.9 Reading In a CSV File

26.1. The Graphics Class
26.1.1. Class Summary
26.1.2. Class Methods
26.1.3. The Radar Class
26.2. The Color Class
26.2.1. Class Summary
26.2.2. Class Usage
26.2.3. Adding Color to the Radar
26.2.4. Building a Color Map
26.2.5. The Color Grid
26.3. The FontMetrics Class
26.3.1. Class Summary
26.3.2. Class Usage
26.3.3. How to Draw a String with a Background
26.3.4. How to Draw a Vertical String, The Target Class
26.4. Charts
26.4.1. The LineGraph class
26.4.2. The BarGraph class
26.4.3. The PieGraph Class
26.5. Images
26.5.1. The ImageUtils class
26.5.2. The ImageFrame class
26.5.3. The WriteGIF class
27. Spiral Components
27.1 The Spiral Class
27.2 Archimedes' Spiral
27.3 Fermats' Spiral Component Class
27.4 Exercises

Grading:
1/3 Homework and Projects
1/3 Midterm
1/3 Final

Computer Usage:
Web and e-mail access are ***REQUIRED.*** You **MUST** have access to a computer with a Java compiler.
References:
The Java Class Libraries by Chan and Lee, Addison Wesley (c) 1997. See http://www.docjava.com for ordering this and other books.
Java Source Book by Ed Anuff, The John Wiley and Sons, Inc., First Edition, (c)1996.
Course Notes, as required, available on-line at http://www.docjava.com or by purchase of copies.
The Java Programming Language, Second Edition, by Ken Arnold and James Gosling, Publisher: Addison-Wesley, ISBN: 0-201-31006-6
Java 1.1 Unleashed, by Morrison, ISBN 1575213613
Core Java 2 Volume 1:Fundamentals by Cay Horstmann,Gary Cornell, ISBN: 0130819336
Java 1.2 Developer's Handbook by Philip Heller Simon Roberts, ISBN: 0782121799
Digital Signal Processing Recipes in Java by D. Lyon and H. Rao, M&T Books, (c) 1998.

Image Processing in Jav *a* by D. Lyon (c) 1999.

Goals:
To learn how to design large-scale software systems using Java. Object orientation and packaging are stressed. Testing and coding of several projects, with a variety of modern techniques.
Coordinator:
Douglas Lyon, Professor of Computer and Software Engineering. *E-mail*: lyon@docjava.com

Assignments will be due at the beginning of class. Assignments handed in after the beginning of class will loose 5 points. Assignments handed in after the end of class will lose 10 points. Late homeworks lose 10 points per day late, weeks ends and holidays included. Missed tests result in a zero unless a written excuse is presented.
More details are available about the class at: http://www.docjava.com

Homework requirements:
Print out a listing of the program. Print out the program input and output. You may need to do this at various levels of detail. Hand in a labeled disk with a printout. Place the disk in a #10 letter envelope and staple the envelope to the printout.
In the event that you *must* miss class, you may hand the homework in via a friend or another student. Should you find that you cannot find anyone in time, you must fax the homework in. Attachments will not be accepted. The fax number for homework is:(203)254-4013 fax. This is the School of Engineering fax machine, so please bring the homework to my attention so that it gets routed to me.

4. ECE 415 Eng. Appl. of Numerical Methods

Instructor: Dr. Jerry Sergent
E-Mail: jsergent@mail.fairfield.edu

Description:

Skills in numerical integration and differentiation, numerical solutions of differential equations, data interpolation, and simulations of processes and systems provide the theoretical basis to proceed to computer graphics, voice and signal processing, and modeling.

Prerequisites: SW 408, Corequisite SW 409

Objectives and Outcomes

No.	Objective	Outcome

1	To understand the constraints, limitations, and potential errors of digital computers used to obtain numerical solutions of mathematical expressions	Students will derive equations and write programs to calculate the errors generated when digital computers are used to obtain numerical solutions of mathematical expressions
2	To develop methods for finding numerical solutions to the roots of equations	Students will derive methods and write programs to determine the roots of equations using the Newton-Raphson and other methods
3	To develop methods for solving sets of simultaneous linear equations	Students will derive methods and write programs to solve sets of simultaneous linear equations using the Gauss-Jordan and other methods
4	To develop mathematical expressions for related data sets	Students will derive methods and write programs to determine mathematical expressions for related data sets
5	To develop methods for numerical differentiation and integration	Students will derive methods and write programs to differentiate and integrate mathematical expressions using Simpson's rules and other methods
6	To develop methods to solve ordinary differential equations	Students will derive methods and write programs to solve ordinary differential equations using the Runge-Kutta and other methods

Textbook: Numerical Methods for Engineers, 4th Edition, Steven Chapra, Raymond P. Canale, McGraw-Hill, 2002,ISBN 0-07-243193-8

Performance Indicators and grading:

The final grade will be based on the homework. The homework will be turned in at the beginning of the class lecture. Students will keep a comprehensive notebook of all problems. All homework will be done in Java.

Homework, Class Participation 100%

Class structure:

Lectures will be the primary source of information. Students are expected to attend every class and to participate in class discussions. Homework assignments will be completed each week and reviewed.

Schedule: Thursday night, 6:30 pm to 9:30 pm

January 22 through May 13, 2004

Schedule

Week	Subjects	Chap.	Problems	Objective
1	**Introduction**	1,2	1.1 – 1.9, 1.15 2.1, 2.5, 2.10, 2.14, 2.15	1
2	**Approximations and Round-off Errors** **Truncation Errors**	3,4	3.6, 3.7, 3.8, 3.10 4.1, 4.2, 4.6, 4.7, 4.8, 4.10, 4.12,	1
3	**Roots of Equations**	5,6	5.1 – 5.12 6.1 – 6.13, 6.22 – 6.24	2
4	**Roots of Equations**	7,8	7.1 – 7.9, 7.12, 7.19 8.18, 8.32, 8.36	2
5	**Solution of Linear algebraic Equations**	9, 10	9.1 – 9.12 10.2 - 10.9	3
6	**Solution of Linear Algebraic Equations**	11, 12	11.1 – 11.10 12.13, 12.14, 12.25, 12.27, 12.28, 12.31	3
7	**Curve Fitting**	17, 18	17.1 – 17-16 18.1 – 18.13	4
8	**Curve Fitting**	19, 20	19.1 – 19-9 20.48, 20.49	4
9	**Numerical Differentiation and Integration**	21	21.1 – 21.20	5
10	**Numerical Differentiation and Integration**	22	22.1 – 22.9	5
11	**Numerical Differentiation and Integration**	23, 24	23.1 – 23.9 24.18, 24.25 – 24.27, 24.41	5
12	**Ordinary Differential Equations**	25	25.1 – 25.11	6
13	**Ordinary Differential Equations**	26	26.1 – 26.8	6
14	**Ordinary Differential Equations**	27, 28	27.1 – 27.12, 27.23 28.21, 28.25, 28.36	6
15	**Final Homework Book Due**			

5. SW 402 Database Concepts

This course focuses on the steps required to build and maintain the database infrastructure for client/server applications. The course covers physical design and implementation of

the database, the use of the database to meet the informational needs of a client/server system and the installation, operation and maintenance of the RDBMS software. Specific topics include SQL (Structured Query Language), Utilities provided by the vendor, the use of an RDBMS, backup and recovery of data, security and controls. Students perform a number of hands-on exercises using an RDBMS running on Windows 2000. Microsoft SQL Server or Oracle is used as the software vehicle for lectures and lab exercises. The course is intended for application programmers and database designers in a client/server environment. Lab included.

Instructor:
Joseph J. Corcoran

E-Mail:
jcorcoran@trilegiant.com
(203) 416-2357 (Work)

Lecture Hours:
6:30 PM - 9:30 PM (includes LAB time)

Prerequisites:
none

Class Requirements:
All homework and labwork is to be submitted on time otherwise penalties will be applied.
Two formal exams (Midterm & Final; 1 project and various LAB Assignments in Oracle)

Course Description:
Focus is on the steps required to build and maintain the database infrastructure for client/server applications. The course covers the physical design and implementation of the database, the use of the database to meet the informational needs of a client/server system, and the installation, operation, and maintenance of RDBMS software. Specific topics include SQL, SQL utilities, the use of an RDBMS, hardware and software tuning for maximum performance, backup and recovery of data, security, and control systems. Students perform a number of hands-on exercises using a RDBMS running on Windows NT. Microsoft SQL Server and ORACLE are used as the software vehicles for lectures and lab exercises. The course is intended for application programmers and database designers in a client/server environment. Lab included.

Course Objectives:

1. The student will (Chapter 1 & 2 in Watson)
   (a) understand the key concepts of data management
   (b) recognize that there are many components of an organization's memory
   (c) understand the problems with existing data management systems
   (d) realize that successful data management requires an integrated

understanding of organizational behavior and information technology.
(e) understand the importance of information to society/organization
(f) be able to describe the characteristics of common information delivery systems

Topics Include: Individual Data Management; Organizational Data Management; Components of Organizational Memory; Problems with DBMS; Information and Organizational Change; Change Information

2. The student will (Chapter 3 in Watson; Chapter 1 in Oracle OCP)
(a) model a single entity
(b) define a single database
(c) write queries for a single table database
(d) write simple queries using SELECT/WHERE/ORDER BY clauses
(e) format data retrieved by a query
(f) differentiate between SQL*Plus commands and SQL statements

Topics Include: The Relational Model; Modeling a single entity database; Creating a single table database; Oracle Datatypes; Operators and Expressions; SQL*Plus Basics

3. The student will (Chapter 4 & 10 in Watson; Chapter 2 in Oracle OCP)
(a) model a one-to-many relationship between two entities
(b) define a database with a one-to-many relationship
(c) write queries for a database with a one-to-many relationship
(d) write queries using single-row functions
(e) write queries using GROUP aggregate functions

Topics Include: Single-Row Functions in SQL (NULLS/Character/Numeric Date/Conversion Functions); GROUP (Multi-row) Functions; Limiting Grouped Data with HAVING; Nesting Functions

4. The student will (Chapter 5 in Watson; Chapter 3 in Oracle OCP)
(a) model a many-to-many relationship between two entities
(b) define a database with a many-to-many relationship
(c) write queries for a database with a many-to-many relationship
(d) write SELECT statements to access data from more than one table using equality and non-equality joins
(e) view data using outer joins
(f) write single-row and multiple-row subqueries

Topics Include: Creating a Relational Database with an m:m relationship; Equality and Non-Equality Joins; Cartesian Joins; Outer Joins; Self-Join; Set Operators; Subqueries

5. The student will (Chapter 6 in Watson; Chapter 4 in Oracle OCP)
(a) model a one-to-one and recursive relationships
(b) define a database with one-to-one and recursive relationships
(c) write queries against one-to-one and recursive relationships

(d) be able to describe each DML statement
(e) Insert/Update/Delete rows in a table
(f) Control transactions with the SET TRANSACTION statement
(g) Create/modify users and to create roles for security
Topics Include: Modeling/Mapping/Querying 1-to-1 and Recursive
Relationships; LOCKING a Table; Transaction Control; Privileges
And Roles; Externally Authenticated User Accounts

6. The student will (Chapter 7 in Watson; Chapter 5 in Oracle OCP)
(a) be able to create a data model of a relational database
(b) be able to create/describe/alter/drop database tables and views
(c) be able to drop/rename/truncate tables
(d) be able to retrieve/insert/update/delete data through a view
Topics Include: Data Modeling; Relationship Descriptors as Identifiers; Managing Tables and Views; Creating from Another Table;
Modifying Table Definitions.

7. The student will (Chapter 8 in Watson; Chapter 6 in Oracle OCP)
(a) be able to understand the process of normalization
(b) be able to distinguish between different normal forms
(c) recognize different data modeling approaches
(d) describe constraints
(e) create and maintain primary key constraints/referential integrity constraints/check constraints/sequences
(f) describe the different types of indexes
Topics Include: Normalization; Common Data Dictionary Views; Procedures and Packages; Triggers and the Data Dictionary

(if time permits)
8. The student will (Chapter 9 in Watson; Chapter 7 in Oracle OCP)
(a) know the structures of the relational model
(b) understand relational algebra commands
(c) be able to determine whether a DBMS is completely relational
(d) be able to declare PL/SQL variables
(e) be able to describe the rules of the nested blocks
(f) be able to identify the uses and types of control structures
Topics Include: Data Structures; Integrity Rules; Manipulation Languages; PL/SQL Block; IF Statement; Nested LOOPS and LABELS

9. The student will (Chapter 11 in Watson; Chapter 8 in Oracle OCP)
(a) be able to recommend a data storage structure
(b) be able to recommend a storage divice
(c) be able to write DML statements in PL/SQL
(d) be able to control transactions
(e) be able to determine the outcome of SQL DML statements

Topics Include: Data Coding Standards; Data Storage Devices; Comparative Analysis; Data Compression; Writing Explicit Cursors

Textbook:

(required)

OCP Oracle81 DBA SQL and PL/SQL Study Guide
Sybex, Inc. by Chip Dawes and Biju Thomas
ISBN 0-7821-2682-0 copyright 2000

(optional)

Data Management: Databases and Organizations
(third edition) by Richard T. Watson
John Wiley & Sons, Inc.
ISBN 0-471-41845-5 copyright 2002

Attendance Policy:

Students are responsible to acquire notes and homework assignments from **classmates** in case of absence. Due dates do not change due to absence.

Grading Policy:

Mid-Term: 33%
Project and Lab Assignments: 33%
Final: 33%

6. SW 410 Enterprise Java

This course will explore server-side Java technologies in a structured manner. Students will be exposed to the latest in Java technologies and API's (JSP's, XML, DOM, Servlets, JDBC, EJB's etc).

3 Credit Hours, Pre-req: SW 409. Offered Fall term annually

Textbook: Java for Programmers, by D. Lyon. Copies are available from the SOE office .
Computer Usage: Students MUST have access to a computer with a Java compiler. E-mail access is required.

Course Notes: Handouts/diskettes/e-mail, web page
Where: Mc 203
Who: Prof. Lyon
Voice Phone: (203)641-6293
Fax: 203-877-4187
Web: http://www.DocJava.com
Topics: The amount of coverage for the following topics may vary.
27. Spiral Components
28. JDBC
29. Network Programming
30. Servlets

31. JSP
32. XML
33. More XML Techniques
34. Bean Properties
35. Introduction to Enterprise Java Beans
36 EJB Container
37 Session Beans
38. Session Bean Deployment
39 Entity Beans
40. EJB Security

Additional topics of exploration may include:
Java Mail, JMS, Container Management,
Session Management, Entity API, JSP Tags,
Introspection, Java Beans Management, WML, XSLT, Java 3D,
JAI, CORBA/IDL, RMI, Advanced GUI's, Multi-media
programming (Java Sound API, JMF, QT4J, etc.) and
other advanced topics.

Educational Objective: To improve student communication skills.
Outcome: Students will write and present materials in a lecture format on a regular basis.

Educational Objective: To prepare students for life-long learning.
Outcome: Student will teach themselves and others about cutting edge technologies.

Educational Objective: To help students make sense of the large and growing number of Java technologies.
Outcome: Students will perform exercises based on each presentation to make sure that they understand the technologies.

Grading Policy:

Homework and Laboratory Trials: 1/3
Midterm Exam : 1/3
Final Exam : 1/3

Assignments are due at the beginning of class. Assignments handed in during class lose 5 points, after class 10 points. Late submittals lose 10 points per day including weekends and holidays. Missing a test results in a zero unless a written excuse is presented.

Homework requirements:
Print out a listing of the program. Print out the program intput and output. You may need to do this at various levels of detail. Hand in a labeled disk with a printout. Place the disk in a #10 letter envelope and staple the envelope to the printout.

7. ECE 470 Embedded Systems

Distributed development, connecting peripherals to networks via Java. Plug-and-play paradigm is used to add services on the fly. Multicast and unicast protocols. Service leasing,
lookup services, remote events, sharing data between distributed processes, distributed transactions.
Interfacing hardware (sensors, robotics, etc..) to the Web.

PreReq: SW409

Text: Core Jini by W. Keith Edwards

Syllabus:

* "Foundations," review of background in OOP.
* philosophical differences between Jini and "classical"
* networked and distributed systems technologies.
* history and motivation of Jini; getting and installing the Jini software.
* Basics of distributed systems.
* "Deployment Scenarios,"
* Run java without Java Virtual Machines (JVMs).
* options available to you if you're designing for Jini.
* Client side CGI control in Java
* Controlling an FTP site
* Controlling a mail server
* writing a mail server

* The Servlet API
* Making cookies
* Storing session state
* formatting HTML in Java
* The Design of HTML forms
* The doPost method
* The doGet method
* Creating a GuestBook
* Persistance and serialization
* Storing serialized objects on the server
* Creating survey forms on the server
* Setting the MIME types
* Sending Binary data on the server side
* State and session management
* Using Hash tables to speed persistance computations

8. ECE 455 Sensor Design and Applications

Fairfield University School of Engineering

Instructor: Dr. Jerry Sergent
E-Mail: jesergent@mail.fairfield.edu
Textbook: Sensor Handbook Sabrie Soloman McGraw-Hill, 1998
ISBN 0-07-059630-1

Course Description:

Sensors of different types are used in a wide range of equipment, especially automated equipment, to detect changes in state and to provide the signals necessary to control various functions. This course covers the design, fabrication, and properties of sensors intended to measure a variety of parameters, such as stress, temperature, differential pressure, and acceleration. Sensors are generally connected to electronics systems that process and distribute the signals. The support electronics must identify the signal, separate it from noise and other interference, and direct it to the appropriate point, and are a critical part of the sensor technology. The design and packaging of the support electronics is discussed in detail.

Prerequisites: Electronic Materials

Objectives and Outcomes

No.	Objective	Outcome
1	To understand the basic principles of sensors	Students will be able to analyze and specify thermal, magnetic, motion, humidity, pressure, proximity sensors, and acceleration sensors
2	To understand the basic principles of microelectromechanical systems (MEMS)	Students will be able to analyze and specify MEMS devices fabricated from micromachined silicon
3	To understand the applications of sensors in manufacturing	Students will be able to specify and utilize sensors in robotic manufacturing applications

Performance Indicators and grading:

Three written exams will be given at approximately equal intervals during the term as outlined in the syllabus. These will be primarily take-home exams requiring the analysis and/or design of fiber optic communications systems

Exam 1, Chapters 1-5	35%
Exam 2, Chapters 6-10	35%
Exam 3, Chapters 1-14, Comprehensive	40%

Class structure:

Lectures will be the primary source of information. Students are expected to attend every class and to participate in class discussions.

Week	Subjects	Chap.	Problems	Objective
1	Introduction	1	Handouts	1
2	Thermal sensors	2	Handouts	1
3	Magnetic sensors	3	Handouts	1
4	Pressure sensors	4	Handouts	1
5	Humidity sensors Exam 1 (Chapters 1-4)	5	Handouts	1
6	Motion sensors	6	Handouts	1
7	Proximity sensors	7	Handouts	1
8	Acceleration sensors	8	Handouts	1
9	MEMS	9	Handouts	2
10	Applications of MEMS Exam 2 (Chapters 5-9)	10	Handouts	2
11	Micromachined silicon devices	11	Handouts	2
12	Applications of sensors	12	Handouts	3
13	Sensors in manufacturing	13	Handouts	3
14	Sensors in robotics	14	Handouts	3
15	Exam 3 (Chapters 1-14)			

9. ECE 465 Nonlinear Control

Description:

The course considers the fundamentals of the analysis of linear electric circuits. The properties of the components that make up electrical circuits, including resistors, capacitors, inductors, voltage sources, and current sources are presented. Both independent and dependent sources are considered. The Kirchhoff current and voltage laws are applied to electrical networks to determine voltages and currents at all points in the network. Circuit behavior is analyzed for DC, AC, and transient excitation. First and second order differential equations with constant coefficients are applied to the transient case to achieve the solution. The properties of dependent sources are studied by analyzing the operational amplifier, a basic building block of electronic circuits. Concepts of transformers and three-phase circuits are introduced. Computer simulation is used to verify the analytical solutions.

Textbook: Nonlinear Systems, Third Edition, Hassan K. Khalil, Prentice Hall, 2002, ISBN 0-13-067389-7

Objectives and Outcomes

No.	Objective	Outcome
1	To understand the properties of second-order control systems	Students will analyze second order control systems to determine performance for a given set of initial conditions
2	To understand the criteria for the stability of nonlinear control systems	Students will determine if a control system is stable for a given set of initial conditions
3	To analyze the effect of small perturbations on the performance and stability of nonlinear control systems	Students will introduce small perturbations in computer models of control systems to determine performance and stability for a given set of initial conditions
4	To understand how the analysis of nonlinear control systems may be performed using linear methods	Students will analyze nonlinear control systems using piecewise linear methods for a given set of initial conditions

Performance Indicators and grading:

Three written exams will be given at approximately equal intervals during the term as outlined in the syllabus. These will be primarily take-home exams requiring the analysis and/or design of nonlinear control systems

Exam 1, Chapters 1-5	35%
Exam 2, Chapters 6-10	35%

Exam 3, Chapters 1-14, Comprehensive 40%

Class structure:

Lectures will be the primary source of information. Students are expected to attend every class and to participate in class discussions.

Wk	Subjects	Chap.	Problems	Objective
1	Introduction Second-order Systems	1 2	Handouts	1
2	Fundamental properties of nonlinear control systems	3	Handouts	1
3	Lyapunov Stability	4	Handouts	1
4	Input-Output Stability	5	Handouts	2
5	Passivity Exam 1 (Chapters 1-5)	6	Handouts	
6	Frequency domain analysis of feedback systems	7	Handouts	2
7	Frequency domain analysis of feedback systems	7	Handouts	2
8	Advanced stability analysis	8	Handouts	2
9	Stability of perturbed systems	9	Handouts	3
10	Perturbation theory and averaging Exam 2 (Chapters 6-10)	10	Handouts	3
11	Singular perturbations	11	Handouts	3
12	Feedback control	12	Handouts	4
13	Feedback linearization	13	Handouts	4
14	Nonlinear design tools	14	Handouts	4
15	Exam 3 (Chapters 1-14)			

10. ECE420 Readings in ECE

This course provides an independent research experience. Students ultimately gain confidence in several areas, allowing them to write better and understand technical papers. Students analyze the results of the research, and communicate the conclusions of the work in an open forum. This course designed to help the student formulate a thesis proposal,

perform literature surveys and learn the finer points of technical writing at the graduate level. A meta-paper is written about the literature in the field. The basics of technical writing and research are emphasized.
The student, with guidance from the adviser, decides on a topic of interest. The student surveys the available resources, including the library, the internet, and other potential sources for journal articles, conference papers, and books on the topic. The information is analyzed and appropriate conclusions are drawn from the results. The student communicates the results in both a written paper and in an open forum.
The topic is expected to be in the student's domain of primary interest and will, in many cases, be preliminary to the Master's thesis.
The instruction has been organized to emphasize methods of the writing and research process. Emphasis is on the processes the writer must consider. The student learns how to state a problem, the techniques of analysis, methods of investigation, and functional organization.
Pre-reqs: 15 credit hours at the graduate level
Credit: 3 cr hrs.
Course Overview: In this course, we will examine the elements, methods and format necessary for producing well-organized, clear and concise thesis proposal.
Instructional Methods: We will use the following instructional methods:

1. In-class assignments
2. Class discussions
3. Workshops
4. Group Work
5. Faculty Lecture
6. Exams

The student will address how the research problem focuses on the application of engineering principles to solve one of societies problems. The paper will address the professional, societal, and ethical context of the work. This includes economic tradeoffs between quality and profit.
Course Objectives and Learning Outcomes:
This course designed to support the framing of a problem that has impact on society
1. The students will learn the principles of economic tradeoffs when selecting a thesis problem.

Expected learning outcomes:

a. Apply project schedules
b. Examine ethical ramifications of the problem
c. Be able to perform an ethical study
d. Learn how to review a paper

2. The student will become proficient in a domain related to some combination of hardware and software.

Expected learning outcomes:

a. Demonstrates the ability to utilize research tools to identify relevant literature in the field.
b. Uses appropriate citations to frame a problem.
c. Learn how to select relevant papers

This course requires substantial research effort and emphasis is placed on good engineering practices.

Outcomes:

1. After the student take this course, they will have thesis proposal, with literature survey. A good problem statement will be a part of the thesis proposal. The proposal will appear on the web for general comment.
2. At least one case study in professional ethics
3. Students will select papers from conferences and journals. Each week, every student will submit a review of a paper, totalling 11 reviews. Reviews are in the style of a review submission to an IEEE transactions. Students are not required to check any mathematical proofs, but they much make a judgement on whether the proofs are clear, convincing and elegant. They are to identify topics of future work.

Rationale: Students learn to write reviews to ensure they have read the papers thoroughly. They are initiated to the process of review writing. Student will learn to eliminate grammatical errors and rhetorical problems. An English instructor (skilled in technical writing) would be helpful here. A technical instructor is needed to make sure that the reviews are technically sound. Each review will contribute to a thesis literature search and help to contribute to a problem proposal. Students must read to find what is most relevant to their particular research agenda and must be critical. Reviews are more than summaries, they contain a list of suggestions for improvement.

Non-native speakers of English must be prepared to seek help from instructors who specialize in teaching English as a second-language.

An SOE masters degree requires a thesis proposal. If the thesis proposal is not available at the end of the Readings in ECE course, the student will take an Incomplete and be blocked from registering for a thesis or a project.

11. ECE 550-551 Master's Thesis I and II

The Master's thesis is intended to be a test of the student's ability to formulate a problem, solve it, and communicate the results. The thesis is carefully supervised by a faculty member on an individual basis. A thesis involves the ability to gather information, examine it critically, think creatively, organize effectively, and write convincingly, it is a project that permits you to demonstrate a great many skills that are basic to both academic and work in industry. The student must also submit a paper for possible inclusion in a refereed journal appropriate to the topic.
Credits: 6-9
Prerequisite – Readings in ECE
Registration for this course requires an approved problem statement by an advisor. Once this is obtained, a thesis schedule is established.
The thesis schedule:
A thesis schedule will be established by the thesis advisor, in collaboration with the student. The student and thesis advisor are required to meet regularly and these meetings are documented. The student submits progress reports, as required by the schedule. These

progress reports are submitted to the thesis advisor and if, unsatisfactory, may be forwarded to a standards committee for advise or action.
Advisee (or advisor) transference is possible, but generally not advantagous.
Outcomes:

1. A thesis
2. A submitted paper

Format
The recommended structural sequence for a thesis is:

1. Title Page
2. Certificate
3. Acknowledgments (if any)
4. Preface (if any)
5. Table of contents
6. List of illustrations and tables (if any)
7. Abstract
8. Introduction (if separate from Chapter 1)
9. Chapters in sequence
10. Appendix or appendices (if any)
11. Bibliography

The title page should contain the thesis title, your name, your degree and the year of submission. The table of contents should be fairly comprehensive, since an index is not included.
Beginning with the first page of the Introduction (or Chapter 1, if there is no separate introduction), the pages should be numbered consecutively using Arabic numerals. Preceding pages, except for the title page, should normally be given lower-case Roman numerals.
You must bind an abstract of no more than 400 words within the thesis. You must also submit an additional three copies of the abstract. No single style of bibliography or referencing is required, but you must be consistent.
Printing guidelines:

1. One-and-a-half (1.5) line spacing is preferred, but double-spacing is acceptable. Single-spacing may be used for appendices and footnotes
2. The thesis must be printed on one side of the paper only
3. The margins on each sheet must be at least 40mm on the left-hand side, 20mm on the right hand side, 30mm at the top and 20mm at the bottom

Presentation and binding
You must prepare three copies of your thesis. Each copy should be in a loose leaf form, in a temporary binder.
After your thesis has been examined, and after you have made any recommended corrections or amendments, the copies of your final thesis must be bound in boards, covered with University press boards and the spine must be embossed with gold lettering in the following way:

1. At the bottom and across - the letters SOE
2. 90mm from the bottom and across - the degree and year of submission, for example:

   MS
   2001
3. Evenly spaced between the statement in 2. and the top of the spine - your initials and surname

4. If the spine of the thesis is too thin to support lettering across, the wording can be embossed along the spine, reading from top to bottom
5. A version of the thesis will be mounted on a web server for public view as both PDF and HTML.

Submission

You must submit three hard copies of your thesis for examination.

With your thesis, you must also submit the following three forms:

1. A certificate of Authorship / Originality signed by you stating that the work is yours and that you haven't submitted it for any other degree
2. A Supervisors Certificate signed by your supervisor(s) stating that your work is ready for examination

You can submit your thesis at any time during the year, however you need to allow about 2 months between the time you submit your thesis and when you can graduate.