

COURSE OUTLINE OF RECORD

| | | | |
|-----------------------|----------|----------------------|--------------------------|
| Dept., Number | CSC 3361 | Course Title | Intermediate Programming |
| Semester Hours | 3 | | |
| Year | 2006 | URL (if any): | |

Current Catalog Description:

This course offers continued development in program design. Larger programs are introduced incorporating string and file processing, internal search/sort methods, and simple data structures. Laboratory work required. Prerequisite: CSC 1310.

Textbook:

Starting Out with Java Early Objects by Tony Gaddis, Scott/Jones, Inc. Publishers, 2004

Course Goals:

To provide computer science students with the skills to:

2. Continue developing a disciplined approach to the design, coding, and testing of programs written in a block-structured high-level language.
2. Teach the use of data abstraction using basic data types such as strings, multi-dimensional, arrays, records, sets, files, and also data structures such as linked lists.
3. Introduce searching and sorting algorithms and their analysis.
4. Provide a foundation for further studies in computer science.

Prerequisites by Topic:

8. Introduction to Computers and Programming
9. Problem solving
10. Basics of object-oriented programming in Java: parts of a program, flow of control, primitive data types and String class, operators
11. Classes and objects, including constructors, overloading, objects as arguments and return values, equals and toString methods.
12. Selection structures: if, if-else, switch
13. Repetition structures: while, do while, for
14. Input and output: interactive and with text files

Major Topics Covered in the Course (number of weeks):

- | | |
|-------------------------------------------------------------------------------------------------------------------|---|
| 1. Classes: Aggregation, this reference variable, packages, inner classes, garbage collection | 2 |
| 2. Arrays: one-dimensional arrays, arrays of strings and objects, searching and sorting, multi-dimensional arrays | 4 |
| 3. Text processing and Wrapper Classes | 2 |
| 4. Inheritance and Polymorphism | 4 |
| 5. Streams and Exception Handling | 3 |
| 6. GUI Applications and Applets | 6 |
| 7. Professional Ethics and Recursion | 2 |

Laboratory Projects:

8. Program using one-dimensional arrays (2 weeks)
9. Program using multidimensional arrays (1 weeks)
10. Program using inheritance and polymorphism (2 weeks)
11. Program using streams and exception handling (2 weeks)
12. Program using GUI applications and applets (3 weeks)
13. Program using recursion (2 weeks)
14. Project on professional ethics

| | | | |
|----------------------|----------|---------------------|--------------------------|
| Dept., Number | CSC 3361 | Course Title | Intermediate Programming |
|----------------------|----------|---------------------|--------------------------|

Estimate Curriculum Category Content (Semester hours)

| Area | Core | Advanced | Area | Core | Advanced |
|-----------------|-------------|-----------------|-----------------|-------------|-----------------|
| Algorithms | 1 | | Data Structures | 1 | |
| Software Design | 1 | | Prog. Languages | | |
| Comp. Arch. | | | | | |

Oral and Written Communication:

Typically, every student is required to submit at least one written report (not including exams, tests, quizzes, or commented programs) of six to ten pages and to make one oral presentation of seven minutes duration. Students normally write summaries of 3-5 computer related articles and present them during lab sessions.

Social and Ethical Issues:

One class period is spent on this topic. Students are assigned relevant topics for papers and are expected to present their findings orally!

Theoretical Content:

Please list the types of theoretical material covered and estimate the time devoted to such coverage.

Problem Analysis:

Problem Solving Techniques. Problem analysis is a vital part of all program development.

Open-Ended Design:

Students are expected to submit algorithms and structure charts with each major programming assignment.