SAULT COLLEGE OF APPLIED ARTS & TECHNOLOGY
SAULT STE. MARIE, ONTARIO

COURSE OUTLINE

Course Title: **DATABASE DESIGN AND IMPLEMENTATION II**

Code No.: **CSD303** Semester: **WINTER 2000**

Program: **PROGRAMMER(2090)/ PROGRAMMER ANALYST(2091)**

Instructor: **DENNIS OCHOSKI**

Date: **JANUARY 2000** Previously Dated: **SEPTEMBER 1999**

Approved: _____ _____
Dean Date

DATABASE DESIGN AND IMPLEMENTATION II — CSD303

COURSE NAME — COURSE CODE

TOTAL CREDITS: 4

PREREQUISITE(S): CSD204

I. COURSE DESCRIPTION:

This course is a continuation of Database Design and Implementation I where more advanced design and implementation of systems will be completed. A major focus of the course is on the physical implementation of a database using a popular platform(s) currently adopted by many organizations. The platforms to be studied this semester will be SQL Server 7.0 and Microsoft Access 97. The course will also extend the concepts of database management to include such topics as client/server architecture, Structured Query Language(SQL), backup and recovery, privacy and security, and, design and implementation of data warehouses.

II. TOPICS TO BE COVERED:

1. Structured Query Language (SQL) using TRANSACT-SQL.
2. Microsoft Access 97: Advanced Implementation.
3. Problems/Solutions in a Multi-User Environment.
4. SQL Server 7.0: Implementation & Administration.
5. Data Warehouses.

III. LEARNING OUTCOMES AND ELEMENTS OF THE PERFORMANCE:

Upon successful completion of this course the student will demonstrate the ability to:

1. Discuss and apply the concepts related to Structured Query Language (SQL) with particular emphasis on TRANSACT-SQL using SQL Server's Query Analyser.
   (Wynkoop: chapters 5-16 and lecture notes)

   This learning outcome will comprise approximately **25%** of the course.

   Elements of the performance:

   A) apply the concepts of database creation by being able to:

   - define and create databases and transaction logs
   - use the TempDB database
   - create and drop backup devices
   - create tables
   - understand and select datatypes
   - define the meaning of a *null* value and *not null* value
   - create and use constraints

   B) apply the concepts of data retrieval by being able to:

   - list the tables in the demonstration database (PUBS)
   - query the database retrieving column and row data
   - query the database using conditions to restrict data retrieved
   - use boolean expressions in the condition of a query
   - use an editor to change the SQL buffer
   - rename column headings in the retrieved data
   - use the numeric functions in data retrieval

Elements of the performance(cont'd):

C) apply the concepts of organizing data and summarizing results by being able to:

- use aggregate functions such as *AVG, COUNT, MIN, MAX,* and *SUM*
- organize data into groups by using the *GROUP BY* clause
- set conditions on groups by using the *HAVING* clause

D) apply the concepts of joins and subqueries by being able to:

- create a join based upon conditions
- create a join which joins a table to itself (self-join)
- create a join that includes non-matching rows (outer join)
- create a join of more than two tables
- create a query with multiple levels
- create a sub-query with comparison operators
- create a sub-query for an existence test

E) apply the concepts of data definition and manipulation by being able to:

- create a table from an existing table
- insert data into a table
- update data in a table
- delete data from a table

F) apply the concepts of data control by being able to:

- control access to the server, a database, commands, and objects
- create and use views to control access
- update tables via views
- create rules and defaults
- apply *triggers* to control updates

Elements of the performance(cont'd):

G) apply the more advanced concepts of SQL Server by being able to:

- describe and use *batches*
- describe and use *stored procedures*
- explain and write transactions
- explain and use backup and restore procedures on a database

2. Create a database in MicroSoft Access and design appropriate applications to process that database.
(lecture notes and previous reference material)

This learning outcome will comprise approximately **25%** of the course.

Elements of the performance:

- define application control mechanism design alternatives
- understand principles of form and report design
- describe the relationship of database structure and application program design
- create a database and design its appropriate tables
- add, edit, and delete records within a table
- describe and apply the data types and properties within Access
- use the Table Wizard to create and update a table
- discuss the importance of data validation and how it is implemented in Access
- use the Table Wizard to create predefined forms
- distinguish between a bound control, an unbound control, and a calculated control; and implement each type in a form
- modify an existing form to include a combo box, command buttons, and colour
- describe the similarities between forms and reports with respect to bound, unbound, and calculated controls
- list the sections that may be present in a report and explain the purpose of each
- differentiate between a query and a table; explain how the objects in an Access database (ie. tables, forms, queries, and reports) interact with one another
- use the Query By Example (QBE) grid to create and modify a select query
- explain the use of multiple criteria rows within the QBE grid to implement And and Or conditions in a query
- use the Relationships window to implement relationships

Elements of the performance(cont'd):

- define referential integrity and explain how its enforcement maintains consistency within a database
- distinguish between a main form and a subform; explain how a subform is used in conjunction with a one-to-many relationship and use the Form Wizard to create a subform
- create a report based on a query
- create a main and a subform based on a query and discuss the advantage of using queries rather than tables as the basis for a form or report
- create a parameter query and explain how it can be made to accept multiple parameters
- use aggregate functions in a select query

3. Understand the problems inherent in a multi-user database environment, and, provide solutions to those problems.
(Kroenke: chapter 12 pgs. 284 - 297, and lecture notes)

This learning outcome will comprise approximately **20%** of the course.

Elements of the performance:

- identify problems caused by concurrent processing
- explain methods to prevent loss of updates and the "deadly embrace"
- define the terms; logical transaction, before image, after image, rollback, and rollforward
- describe the problems related to database recovery
- explain methods for recovery after certain types of system failures
- describe the problems associated with database security and how database management software handles security implementation
- explain object-oriented and subject-oriented security.
- understand client-server database architecture and know the components of client-server systems and the role of each component
- describe the characteristics of other various multi-user processing architectures such as centralized, shared, and file-server
- understand why client-server systems have advantages over other architectures

4. Acquire an understanding of Microsoft SQL Server 4.0.
(Wynkoop: chapters 1- 4)

This learning outcome will comprise approximately **20%** of the course.

Elements of the performance:

- understand the evolution of SQL Server and its components, including the installation and configuration of a database and its client components
- understand the N-Tier features of SQL Server
- understand the concepts of multiprocessing, multitasking, multithreading, and multiarchitecture
- understand the Windows NT network components, workgroups, and domains
- understand the NT performance monitor
- install and set up the Server and Client software as well as the tools from the client system
- apply the concepts in outcome #3 to SQL Server 7.0.

5. Understand how a data warehouse is used to provide decision-support personnel with historical data needed for trend analysis.
(Kroenke: chapter 15 pgs. 381 - 388)

This learning outcome will comprise approximately **10%** of the course.

Elements of the performance:

- describe the purpose and structure of a data warehouse
- differentiate between "snapshot" data and "ongoing" data with respect to the operational environment vs the data warehouse environment
- understand how data is transferred from the operational environment to the data warehouse
- understand the design and implementation of a data warehouse
- differentiate between a "data warehouse" and a "data mart"

IV. EVALUATION METHODS:

The mark for this course will be arrived at as follows:

Quizzes:		
	outcome #1	15%
	outcome #2	15%
	outcome #3	15%
	outcome #4 & #5	20%
Assignments		10%
Projects		25%
	Total	100%

The grading scheme used will be as follows:

A+	90 - 100%	Outstanding achievement
A	80 - 89%	Excellent achievement
B	70 - 79%	Average achievement
C	60 - 69%	Satisfactory achievement
R	< 60%`Repeat the course	
X	Incomplete	A temporary grade limited to special circumstances that have prevented the student from completing the objectives by the end of the semester. An X grade reverts to an R grade if not upgraded within a specified time period.

V. ASSIGNMENT/PROJECT SPECIFIC INFORMATION

1. Projects will be assigned to student "project teams", each consisting of two or three students.

2. It is the responsibility of the project team to clarify any system requirements with the instructor.

3. At various intervals, the instructor will require each project team to report on the progress made on their respective project. At that time, each team member will be required to complete a Peer Evaluation Form used to "grade" each team member's contribution to the project.

4. At the completion of a project, the respective project team will present and demonstrate the functionality of their system to the instructor.

5. The grade assigned to the overall project and to each team member will be determined using these sources:

    a) Peer Evaluation Form
    b) Presentation of project to instructor(s)

    ** Note: When a project is presented to the instructor, each team member will be required to demonstrate his/her assigned task(s). The project will receive an overall grade and each team member will receive an individual grade which may or may not be equivalent to the overall project grade or to the grades of other team members.

VI. SPECIAL NOTES

1. In order to pass this course the student must obtain an overall **quiz** average of 60% or better, as well as, an overall **assignment** average of 60% or better. A student who is not present to write a particular quiz, and does not notify the instructor beforehand of their intended absence, may be subject to a zero grade on that quiz.

2. There will be **no** supplemental or make-up quizzes/tests at the end of the semester.

3. Assignments must be submitted by the due date according to the specifications of the instructor. Late assignments will normally be given a mark of zero. Late assignments will only be marked at the discretion of the instructor in cases where there were extenuating circumstances.

4. Any assignment submissions deemed to be copied will result in a **zero** grade being assigned to **all** students involved in that particular incident.

5. Students with special needs (eg. physical limitations, visual impairments, hearing impairments, learning disabilities) are encouraged to discuss required accommodations confidentially with the instructor.

6. The instructor reserves the right to modify the course outcomes and/or the assessment process to meet any changing needs of the class.

VII. PRIOR LEARNING ASSESSMENT:

Students who wish to apply for advanced credit in the course should consult the instructor.

VIII. REQUIRED STUDENT RESOURCES

Texts: DATABASE PROCESSING: Fundamentals - Design - Implementation, 7th edition,
by David Kroenke
Prentice Hall Publishing

Using Microsoft Server 7.0: Special Edition,
by Stephen Wynkoop
Prentice Hall Publishing