SAULT COLLEGE OF APPLIED ARTS AND TECHNOLOGY

SAULT STE. MARIE, ONTARIO

COURSE OUTLINE

COURSE TITLE	COBOL I
CODE NUMBER	EDP112
PROGRAM	COMPUTER PROGRAMMER
SEMESTER	THREE
DATE	SEPTEMBER, 1993
AUTHOR	FRAN DEW

NEW ___ REVISION X

APPROVED Rose M. Lacroix 93/09/01

CHAIRPERSON DATE

COBOL I
COURSE NAME

EDP112
COURSE CODE

Total credit time: 90 hours

Prerequisites: EDP111, EDP318

I PHILOSOPHY/GOALS:

This course provides the student with an opportunity to develop practical data processing skills through structured COBOL programming. This course also provides an opportunity for the student to develop skills in using standard techniques for problem analysis through to final program testing.

II STUDENT PERFORMANCE OBJECTIVES:

Upon successful completion of this course, the student will be able to:

1. understand the origin, purpose and basic structure of COBOL, as well as understand its general coding conventions and format rules
2. design structured programs
3. write high-level cobol programs
4. maintain files
5. use advanced COBOL features

III TOPICS TO BE COVERED

1. Introduction to structured program design
2. IDENTIFICATION and ENVIRONMENT DIVISIONS
3. DATA DIVISION - File Section, Working-Storage Section
4. PROCEDURE DIVISION
5. The theory of structured program design
6. Moving data and printing information
7. Debugging programs
8. Computing in COBOL
9. Selection using the IF statement
10. Iteration
11. Control break processing
12. Sequential file processing
13. Sorting
14. The COPY statement

IV LEARNING ACTIVITIES

1. Chapter 1 Introduction to Structured Program Design

   Upon successful completion of this unit, the student will be able to:
   a describe COBOL as a business-oriented language
   b relate COBOL programming techniques and practices
   c describe an overview of the four divisions of COBOL

2. Chapter 2 IDENTIFICATION and ENVIRONMENT DIVISIONS

   Upon successful completion of this unit, the student will be able to:
   a describe the basic structure of a COBOL program
   b employ the general coding and format rules
   c use IDENTIFICATION DIVISION and ENVIRONMENT DIVISION entries

3. Chapter 3 DATA DIVISION - File Section, Working-Storage Section

   Upon successful completion of this unit, the student will be able to:
   a use system design, relating to programming
   b organize data, and form data-names and constants
   c define and describe input and output files in the DATA DIVISION
   d reserve storage for constants and work areas

4. Chapter 4 PROCEDURE DIVISION

   Upon successful completion of this unit, the student will be able to:
   a access input and output files, read data from an input file and write information onto an output file
   b perform move statements
   c execute paragraphs from a main module, and perform end-of-job operations

5. Chapter 5 The Theory of Structured Program Design

Upon successful completion of this unit, the student will be able to:

a map out structured program logic using flowcharts and pseudocode
b illustrate the relationships among modules in a top-down program using hierarchy or structure charts
c use the logical control structures of sequence, selection, iteration and case
d use techniques to make programs easy to code, debug, maintain, and modify

6. Chapter 6 Moving Data and Printing Information

Upon successful completion of this unit, the student will be able to:

a use options of the MOVE statement
b describe the rules for moving fields and literals
c print decimal points and dollar signs

7. Chapter 11 Debugging programs

Upon successful completion of this unit, the student will be able to:

a anticipate the types of input errors that might occur
b use techniques to validate input data
c perform actions upon error detection

8. Chapter 7 Computing in COBOL

Upon successful completion of this unit, the student will be able to:

a perform arithmetic in COBOL in various ways
b use the formats and options available with the arithmetic verbs

9. Chapter 8 Selection using the IF statement

Upon successful completion of this unit, the student will be able to:

a use the IF statement for selections
b use a variety of formats and options available with the conditional statement

10. Chapter 9 Iteration

Upon successful completion of this unit, the student will be able to:

a use the PERFORM statement options for iteration

11. Chapter 10 Control Break Processing

Upon successful completion of this unit, the student will be able to:

a prepare the main types of computer-generated reports, using techniques for efficient printing of group reports and control totals

b use control break processing and printing

12. Chapter 14 Sequential File Processing

Upon successful completion of this unit, the student will be able to:

a update sequential master files using various techniques

13. Chapter 15 (part) Sorting

Upon successful completion of this unit, the student will be able to:

a process a file before, during and after it is sorted

14. Chapter 19 (part) The COPY statement

Upon successful completion of this unit, the student will be able to:

a copy program segments from a computer library

COURSE NAME COURSE CODE

V EVALUATION METHODS

Tests (3 @ 25%)	75%
Assignments (5 @ 5%)	25%
	100%

Assignments received after the due date are subject to a zero mark. Assignment layout is explained on another handout.

Grading:		
	A+	90 and over
	A	80 and over
	B	70 and over
	C	60 and over
	R	under 60

VI REQUIRED STUDENT RESOURCES

Text: "Structured COBOL Programming"
by Stern & Stern
6th Edition, 1991
available in the Campus Bookstore

Other References: VAX COBOL Volume 1 User Manual
VAX COBOL Volume 2 Reference Manual
available in Software Support

VII SPECIAL NOTES

Tests may contain both written and practical on-line components.

Visual impairments, hearing impairments, or learning disabilities, are encouraged to discuss required accommodations, confidentially, with the instructor.

Your instructor reserves the right to modify the course as she deems necessary to meet the needs of students.

EDP112 COBOL I ASSIGNMENT 1

At the SCAAT BOUTIQUE, a monthly ACCOUNTS-RECEIVABLE report is to be created from a customer data file. All customer record information is to be printed on the report, along with the account service charge and amount due, which are calculated from the record information. Service charge is calculated at 1.5% of this month's balance. Grand totals of the a)month's sales, b)payments and c)amount due are also to be printed at the end of the report.

An appropriate heading is to be put on each page of the report, along with page number and date of run. Print a maximum of 20 detail lines on each page.

If the BALANCE FROM LAST MONTH is greater than the PAYMENT ON ACCOUNT,

SERVICE CHARGE =
0.015 * (BALANCE FROM LAST MONTH - PAYMENT ON ACCOUNT)

If the BALANCE FROM LAST MONTH is not greater than the PAYMENT ON ACCOUNT,

SERVICE CHARGE = 0.

The formula for AMOUNT DUE is:

BALANCE FROM LAST MONTH - PAYMENT ON ACCOUNT

+ SERVICE CHARGE + SALES FROM THIS MONTH

NOTE:

BALANCE FROM LAST MONTH might be negative to reflect a credit to the account
SALES FROM THIS MONTH might be negative to reflect possible return of merchandise
AMOUNT DUE might be negative to reflect a credit

DATA

Set up a customer data file of at least 50 records filled with data you provide. When you set up the data file, put negative values in some records, to test the negative possibilities. Reference text, page 250, for information about entering negative numbers in a data file.

The layout of the input record is as follows:

CUSTOMER NAME	15 columns	
CUSTOMER ADDRESS	35 columns	
ACCOUNT NUMBER	5 columns	
BALANCE FROM LAST MONTH	5 columns	use S999V99
SALES FROM THIS MONTH	5 columns	use S999V99
PAYMENT ON ACCOUNT	5 columns	

EDP112 COBOL I ASSIGNMENT 2

At SCAAT BOUTIQUE, a WEEKLY WAGE report is to be created from an employee data file. Employees are paid either on an hourly or a commission basis. Records in error are to be listed in a WEEKLY WAGE DATA ERROR report, with the error listed beside the record.

HOURLY EMPLOYEES
Hourly employees are paid on a 40 hour work week, with overtime paid at time and a half. The allowable rate is in the range of $6.50 per hour to $15.00 per hour.

COMMISSION EMPLOYEES
Commission employees are paid as follows:

For Weekly Gross Sales to and including $5,000:
10% of weekly Gross Sales

For Weekly Gross Sales over $5,000:
10% of weekly Gross Sales to and including $5,000, plus
15% on weekly Gross Sales over $5,000

INCOME TAX
Income tax is calculated at 35% of gross wage.

PRINT OUTS
For each employee, a printout of the input is required, plus Gross regular pay, Gross overtime pay, 10% and 15% commissions, total gross wage, income tax, and net wage for each employee.

Overall totals for all money columns are also required.

DATA VALIDATION CHECKS
Perform DATA VALIDATION CHECKS on the data, <u>including</u> the following:

check for an employee number
check for an employee name
check for a rate of pay, if there are hours worked
make sure that hours worked and sales don't both have values in the same record
check for minutes over 59

DATA
Have a minimum of ten records in the test data. The layout of the input record is as follows:

EMPLOYEE NUMBER	5 columns
EMPLOYEE NAME	15 columns
HOURS WORKED	2 columns
MINUTES WORKED	2 columns
RATE OF PAY	4 columns
SALES	6 columns (two decimal places)

EDP112 COBOL I ASSIGNMENT 3

At the SCAAT BOUTIQUE, a PAST-DUE ACCOUNTS (also called an Aged Trial Balance) report is to be created from a customer data file. On each output line, list customer number, customer name, original balance due, the discount or service charge (this will be in one of ~~three~~ two columns - refer to the information in "If not a credit"), updated balance due, the control code value and a written description of the control code meaning. Records in error are to be listed in a PAST-DUE ACCOUNTS DATA ERROR report, with the error listed beside the record.

DATA VALIDATION CHECKS
Perform DATA VALIDATION CHECKS on the data, including the following:

Check for CONTROL CODE not 1,2 or 3
Check for non-numeric CUSTOMER NUMBER
Check ORIGINAL BALANCE DUE for field left blank, or with leading zeros left off

Totals are to be printed for each money column.

IF NOT A CREDIT:
If the account is current (0 to 30 days), a 3% cash discount is given.
If the account is past due (31 to 60 days), no cash discount is given.
If the account is over 60 days past due, a 1.5% service charge is added to the balance due.

DATA
Have a minimum of 15 records of test data. The layout of the input record is as follows:

CUSTOMER NUMBER	5 columns
CUSTOMER NAME	15 columns
ORIGINAL BALANCE DUE	5 columns (may be a credit)(two decimal places)
CONTROL CODE	1 column where value is one of: 1=0-30 days (current account) 2=31-60 days past due 3=61 days and over past due

EDP112 COBOL I ASSIGNMENT 4

At the SCAAT CREDIT UNION, a group-indicated BANK ACCOUNT BALANCE report is to be created from an account data file. Individual transactions are to be printed, and a balance for each account is to be accumulated and printed. A final over-all total for balances is to be output at the end of the report. Records in error are to be listed in a BANK ACCOUNT BALANCE DATA ERROR report, with the error listed beside the record.

Data validation is to be performed on all numeric fields.

Sort the records in ascending order by date within account number. "Group-indicated" means that the account number will be printed on the first line of each account output, but not on the succeeding lines for that account.

DATA

Have a minimum of 15 records of test data. The layout of the input record is as follows:

ACCOUNT NUMBER	5 columns
TRANSACTION DATE	6 columns in the form DDMMYY
AMOUNT	8 columns (two decimal places)
TRANSACTION CODE	1 column

where transaction code is: 1=old balance
2=deposit
3=withdrawal

Hint: Sort day within month, within year, within account number

EDP112 COBOL I ASSIGNMENT 5

For the SCAAT BOUTIQUE, create a NEW MASTER customer data file containing an OLD MASTER customer data file's information which has been updated by a TRANSACTION customer data file. Print the resulting information in a CUSTOMER UPDATE report, indicating what, if any, modification was done on each account. List all deleted records in a CUSTOMER DELETION report.

MASTER records in error are to be listed in a MASTER RECORD ERROR report, with the error listed beside the record. TRANSACTION records in error are to be listed in a TRANSACTION RECORD ERROR report, with the error listed beside the record.

Some steps to follow are:

Sort the OLD MASTER customer data file on account number
Sort the TRANSACTION customer data file on account number
Perform data validation on the Old Master and Transaction files.
Update the Old Master file with the Transaction file, following the logic of the diagram on the following page.

DATA
Have a minimum of 15 records of test data, in each of the input files.

The layout of the Old Master and New Master record is as follows:

CUSTOMER NAME	15 columns
CUSTOMER ADDRESS	35 columns
ACCOUNT NUMBER	5 columns
FILLER	15 columns

The layout of the Transaction record is as follows:

CUSTOMER NAME	15 columns
CUSTOMER ADDRESS	35 columns
ACCOUNT NUMBER	5 columns
TRANSACTION CODE	1 column

where transaction code is:

A	addition	-	add a new customer
C	change	-	change customer name and/or address
D	delete	-	delete old customer

EDP112 COBOL I ASSIGNMENT 5 UPDATING DIAGRAM

UPDATE AN OLD MASTER FILE WITH A TRANSACTION FILE

ACCOUNT NUMBER ON TR VERSUS ACCOUNT ON OM

	TR < OM	TR = OM	TR > OM
A	1 add TR to NM read TR	2 ERROR read TR	3 write OM to NM read OM
C	4 ERROR read TR	5 write TR to NM read OM read TR	6 write OM to NM read OM
D	7 ERROR read TR	8 delete record read OM read TR	9 write OM to NM read OM

OM = Old Master record

TR = Transaction record

NM = New Master record

(i.e. for box 4, if the customer number of the Transaction record is less than that of the Old Master record, then there is an error, because there must be a corresponding customer in the Old Master file for a change to be made.)