COBOL Programming I

Course Design

2000-01

Course Information

Organization:	Eastern Arizona College
Division:	Business-Computer Department
Course Number:	CMP 221
Title:	COBOL Programming I
Credits:	3
Developed by:	Ed Hunter
Lecture/Lab Ratio:	Lect 2/Lab 2
Transfer Status:	NAU, ASU, UA
Extended Registration Class:	N/A
CIP Code:	11.0201
Assessment Mode:	Pre/Post Test, 25 Questions, 100 Points
Awareness Course:	No
Intensive Writing Course:	No
Prerequisites:	1. No prerequisite.
Goals:	1. Prepare the student at the entry level for vocational programming in the COBOL language.
Description:	A comprehensive introduction to the COBOL language; preparation and writing of programs in the COBOL language, using current programming techniques. A previous programming language is helpful.
Textbooks:	Shelly/Cashman/Forsythe. *Structured COBOL Pseudocode Edition.* Latest. Boyd and Fraser Publishing Company, Latest. This textbook is required. Source: Bookstore. Cost: $50.00.
Supplies:	3 1/2 inch HD diskette

Competencies and Perfomance Standards

1. Describe process for preparing a computer program.

Domain--Cognitive *Level--Evaluation* *Importance--Essential* *Difficulty--Low*

Criteria--Criteria - Performance will be satisfactory when:	**Conditions**--Competence will be demonstrated:	**Learning Objectives:**
• learner hears/reads problem scenario. • learner defines problem. • learner uses ALGORITHM tools for program planning. • learner writes code using ALGORITHM. • learner keys in code to create source program file. • learner compiles source program into object code. • learner runs program. • learner debugs program. • learner jumps up and down for joy.	• by preparing a required first trial program. • by writing future programs for course.	a. Hear/read problem scenario. b. Define problem. c. Use ALGORITHM tools for program planning. d. Write code using ALGORITHM. e. Key in code to create source program file. f. Compile source program into object code. g. Run program. h. Debug program. i. Jump up and down for joy.

2. Create algorithms for the development of a COBOL program.

Domain--Cognitive *Level--Analysis* *Importance--Essential* *Difficulty--High*

Criteria--Criteria - Performance will be satisfactory when:	**Conditions**--Competence will be demonstrated:	**Learning Objectives:**
• learner hears/reads problem definition. • learner applies PSEUDOCODE and/or FLOWCHARTING to develop algorithm. • learner uses resulting algorithm to code program.	• by developing algorithm using algorithm tool. • by successful completion of final exam.	a. Read or hear computer programming problem description. b. Define problem for computer programming problem. c. Use algorithm tool to sequence problem solving steps.

3. Write code in the COBOL programming language.

Domain--Cognitive *Level--Application* *Importance--Essential* *Difficulty--High*

Criteria--Criteria - Performance will be satisfactory when:	**Conditions**--Competence will be demonstrated:	**Learning Objectives:**
• learner uses algorithm to develop code. • learner uses CODING form to code program. • learner keys code to program file. • learner compiles program to determine error conditions. • learner writes correct code to repeat debug cycle until program is correct.	• by completion of successful coded program throughout each chapter. • by successful completion of required program. • by successful completion of final exam.	a. Use algorithm to develop code. b. Use CODING form to code program. c. Key code to program file. d. Compile program to determine error conditions. e. Write correct code to repeat debug cycle until program is correct.

4. Use INPUT/OUTPUT commands in a COBOL program when and where appropriate.

Domain--Cognitive *Level--Analysis* *Importance--Essential* *Difficulty--Medium*

Criteria--Criteria - Performance will be satisfactory when:

- learner uses CODING form to code program.
- learner prepares the I/O select statements in the ENVIRONMENT DIVISION.
- learner prepares the FD in the DATA DIVISION.
- learner prepares I/O sections of WORKING-STORAGE SECTION.
- learner uses LABEL RECORDS clause and RECORD CONTAINS clause in FD.
- learner distinguishes Level Numbers in file organization.
- learner applies appropriate Picture Clauses in DATA DIVISION.
- learner applies OPEN and CLOSE statements in PROCEDURE DIVISION.
- learner applies MOVE statements to establish OUTPUT.
- learner uses READ and WRITE command for I/O.
- learner uses VALUE CLAUSE for data definition.
- learner uses PERFORM statement in appropriate sequence of COBOL program.
- learner uses STOP RUN statement.
- learner uses literals for data definition.
- learner desk-runs sample program illustrating I/O commands until understood.
- learner participates in lecture/ demo illustrating I/O commands.
- learner applies I/O commands to appropriate place in program logic while coding COBOL program.

Conditions--Competence will be demonstrated:

- by applying I/O commands to COBOL program which runs successfully.
- by successful completion of final exam.

Learning Objectives:

a. Use CODING form to code program.
b. Prepare the I/O select statements in the ENVIRONMENT DIVISION.
c. Prepare the FD in the DATA DIVISION.
d. Prepare I/O sections of WORKING-STORAGE SECTION.
e. Use LABEL RECORDS clause and RECORD CONTAINS clause in FD.
f. Distinguish Level Numbers in file organization.
g. Apply appropriate Picture Clauses in DATA DIVISION.
h. Apply OPEN and CLOSE statements in PROCEDURE DIVISION.
i. Apply MOVE statements to establish OUTPUT.
j. Use READ and WRITE command for I/O.
k. Use VALUE CLAUSE for data definition.
l. Use PERFORM statement in appropriate sequence of COBOL program.
m. Use STOP RUN statement.
n. Use literals for data definition.
o. Desk-run sample program illustrating I/O commands until understood.
p. Apply I/O commands to appropriate place in program logic while coding COBOL program.

5. Use ARITHMETIC OPERATIONS in a COBOL program when and where appropriate.

***Domain**--Cognitive* ***Level**--Analysis* ***Importance**--Essential* ***Difficulty**--Medium*

Criteria--Criteria - Performance will be satisfactory when:	**Conditions**--Competence will be demonstrated:	**Learning Objectives:**
• learner applies ROUNDING functions to arithmetic results. • learner uses forms of ADDITION AND SUBTRACTION commands. • learner applies DIVISION forms of arithmetic code. • learner applies MULTIPLICATION forms of arithmetic code. • learner uses the COMPUTE statement. • learner uses different arithmetic editing formats for numeric data, to include zero suppression and insertion characters. • learner defines program constants. • learner prioritizes arithmetic operations correctly. • learner applies TOTAL-LINES in working-storage and output areas. • learner applies ARITHMETIC OPERATION commands to appropriate place in program logic while coding COBOL program.	• by applying ARITHMETIC OPERATION commands to COBOL program which runs successfully. • by successful completion of final exam.	a. Apply ROUNDING functions to arithmetic results. b. Use forms of ADDITION AND SUBTRACTION commands. c. Apply DIVISION forms of arithmetic code. d. Apply MULTIPLICATION forms of arithmetic code. e. Use the COMPUTE statement. f. Use different arithmetic editing formats for numeric data, to include zero suppression and insertion characters. g. Define program constants. h. Prioritize arithmetic operations correctly. i. Apply TOTAL-LINES in working-storage and output areas. j. Apply ARITHMETIC OPERATION commands to appropriate place in program logic while coding COBOL program.

6. Use COMPARING commands in a COBOL program including nested versions when and where appropriate.

Domain--Cognitive *Level--Analysis* *Importance--Essential* *Difficulty--High*

Criteria--Criteria - Performance will be satisfactory when:	**Conditions**--Competence will be demonstrated:	**Learning Objectives:**
• learner applies COMPARING commands to appropriate place in program logic while coding COBOL program. • learner uses forms of the IF-THEN-ELSE structure to include nested versions. • learner uses the relation test. • learner compares numeric items. • learner compares alphanumeric items. • learner uses complex IF statements. • learner uses condition names. • learner applies logical operators.	• by applying COMPARING commands to COBOL program which runs successfully. • by successful completion of final exam.	a. Apply COMPARING commands to appropriate place in program logic while coding COBOL program. b. Use forms of the IF-THEN-ELSE structure. c. Use the relation test. d. Compare numeric items. e. Compare alphanumeric items. f. Use complex IF statements. g. Use condition names. h. Apply logical operators. i. Apply COMPARING commands to appropriate place in program logic while coding COBOL program.

7. Write COBOL code to perform FINAL TOTALS.

Domain--Cognitive *Level--Analysis* *Importance--Essential* *Difficulty--High*

Criteria--Criteria - Performance will be satisfactory when:	**Conditions**--Competence will be demonstrated:	**Learning Objectives:**
• learner defines TOTAL ACCUMULATORS in WORKING-STORAGE area. • learner defines TOTAL variables in output area. • learner develops TOTAL paragraphs in the PROCEDURE DIVISION for required arithmetic commands. • learner applies FINAL TOTAL procedure to appropriate place in program logic while coding COBOL program.	• by applying FINAL TOTAL procedure to COBOL program which runs successfully. • by successful completion of final exam.	a. define TOTAL ACCUMULATORS in WORKING-STORAGE area. b. define TOTAL variables in output area. c. develop TOTAL paragraphs in the PROCEDURE DIVISION for required arithmetic commands. d. apply FINAL TOTAL procedure to appropriate place in program logic while coding COBOL program.

8. Apply REPORT and DATA EDITING when and where appropriate.

Domain--Cognitive *Level--Application* *Importance--Essential* *Difficulty--Medium*

Criteria--Criteria - Performance will be satisfactory when:	**Conditions**--Competence will be demonstrated:	**Learning Objectives:**
• learner applies editing for input fields code. • learner applies checking for blank fields and missing data code. • learner applies testing for reasonableness in an a numeric field. • learner applies testing for specific values in a field. • learner applies the redefines clause. • learner defines the heading line. • learner defines the date work area. • learner applies REPORT and DATA EDITING commands to appropriate place in program logic while coding COBOL program.	• by applying REPORT and DATA EDITING commands to COBOL program which runs successfully. • by successful completion of final exam.	a. Apply editing for input fields code. b. Apply checking for blank fields and missing data code. c. Apply testing for reasonableness in an a numeric field. d. Apply testing for specific values in a field. e. Apply the redefines clause. f. Define the heading line. g. Define the date work area. h. Apply REPORT and DATA EDITING commands to appropriate place in program logic while coding COBOL program.

9. Write COBOL code to perform CONTROL BREAKS including multiple level.

Domain--Cognitive *Level--Application* *Importance--Essential* *Difficulty--High*

Criteria--Criteria - Performance will be satisfactory when:	**Conditions**--Competence will be demonstrated:	**Learning Objectives:**
• learner applies detail record processing. • learner applies summary record processing with group indication. • learner applies FINAL TOTAL processing with CONTROL BREAKS. • learner applies TABLE PROCESSING with CONTROL BREAKS. • learner applies CONTROL BREAKS including multiple level commands to appropriate place in program logic while coding COBOL program.	• by applying CONTROL BREAKS including multiple level commands to COBOL program which runs successfully. • by successful completion of final exam.	a. Apply detail record processing. b. Apply summary record processing with group indication. c. Apply FINAL TOTAL processing with CONTROL BREAKS. d. Apply TABLE PROCESSING with CONTROL BREAKS. e. Apply CONTROL BREAKS including multiple level commands to appropriate place in program logic while coding COBOL program.

10.Write COBOL code for TABLE PROCESSING including external tables.

Domain--Cognitive *Level--Analysis* *Importance--Essential* *Difficulty--High*

Criteria--Criteria - Performance will be satisfactory when:	**Conditions**--Competence will be demonstrated:	**Learning Objectives:**
• learner defines tables within DATA DIVISION. • learner uses subscripting to reference table data. • learner applies perform statement in referencing table data. • learner applies TABLE PROCESSING including external tables commands to appropriate place in program logic while coding COBOL program.	• by applying TABLE PROCESSING including external tables commands to COBOL program which runs successfully. • by successful completion of final exam.	a. Define tables within DATA DIVISION. b. Use subscripting to reference table data. c. Apply perform statement in referencing table data. d. Apply TABLE PROCESSING including external tables commands to appropriate place in program logic while coding COBOL program.

11.Write COBOL code to enable TABLE SEARCHING.

Domain--Cognitive *Level--Analysis* *Importance--Essential* *Difficulty--Medium*

Criteria--Criteria - Performance will be satisfactory when:	**Conditions**--Competence will be demonstrated:	**Learning Objectives:**
• learner applies TABLE SEARCHING commands to appropriate place in program logic while coding COBOL program. • learner writes TABLE SEARCHING code.	• by applying TABLE SEARCHING commands to COBOL program which runs successfully. • by successful completion of final exam.	a. Write TABLE SEARCHING code. b. Apply TABLE SEARCHING commands to appropriate place in program logic while coding COBOL program.

12.Write COBOL code to enable SORTING.

Domain--Cognitive *Level--Analysis* *Importance--Essential* *Difficulty--Low*

Criteria--Criteria - Performance will be satisfactory when:	**Conditions**--Competence will be demonstrated:	**Learning Objectives:**
• learner applies SORTING commands to appropriate place in program logic while coding COBOL program. • learner writes TABLE SEARCHING code.	• by applying SORTING commands to COBOL program which runs successfully. • by successful completion of final exam.	a. Apply SORTING commands to appropriate place in program logic while coding COBOL program. b. Write SORTING code.

13.Compile a COBOL program.

Domain--Cognitive *Level--Application* *Importance--Essential* *Difficulty--Low*

Criteria--Criteria - Performance will be satisfactory when:	**Conditions**--Competence will be demonstrated:	**Learning Objectives:**
• learner performs compilation procedure based upon hardware/ software system available	• by successful compilation of source code into execution code.	a. Perform compilation procedure based upon hardware/software system available

14.Debug a COBOL program.		
Domain--Cognitive *Level--Analysis* *Importance--Essential* *Difficulty--High*		
Criteria--Criteria - Performance will be satisfactory when: • learner prints debug listing resulting from error compilation of program. • learner reads error conditions resulting from error compilation of program. • learner develops strategy for correction and keys in corrected code. • learner reruns program until error free.	**Conditions--**Competence will be demonstrated: • by removing all errors from program assignments during the semester. • by successful completion of final exam.	**Learning Objectives:** a. Print debug listing resulting from error compilation of program. b. Read error conditions resulting from error compilation of program. c. Develop strategy for correction and keys in corrected code. d. Rerun program until error free.

Types of Instruction

Classroom Presentation

On Campus Laboratory and Clinicals

Grading Policy

Evaluation Methods: A programming class evaluates the student on the student's ability to write programs successfully. The grade, therefore, should emphasize that fact. In the current text, through chapter 6 represents a C, through chapter 8 represents a B, and chapter 10 (skipping 9) represents an A.

The Post Test = 10% of the grade.

Grading Scale:

Grade	Requirement
A	90-100%
B	80-89%
C	70-79%
D	60-69%

Learning Plans

Learning Plan 1-- COBOL program basic concepts before programming

Overview: The items in this lesson plan are concepts which you need to have before you start to write and run your programs. They typically do not require using the computer, except in the case of a sample program to show the concepts.

Competency: 1. **Describe process for preparing a computer program.**

Competency: 2. **Create algorithms for the development of a COBOL program.**

Competency: 13. **Compile a COBOL program.**

Competency: 14. **Debug a COBOL program.**

Learning Activities:

_____1. READ the chapter introducing the concept to be learned.

_____2. PARTICIPATE in the lecture/demo presented by the instructor in class.

_____3. LIST concept procedural elements in preparation for using during the writing and execution of a program.

Performance Assessment Activities:

_____1. List concept elements for instructor.

_____2. Apply concepts to required programs which follow during the semester.

_____3. Successfully complete quizzes and final exam for evaluation.

Learning Plan 2-- Standard Lesson Plan for all Units

Overview: All lessons for programming courses follow the process of show and do. You will read the material prior to your instructor's lecture/demo. Key will be the sample program used in the text book which will be covered by your instructor. It is important you understand it before you attempt your own program. Once you understand the chapter, then you should write your own COBOL program and turn in your listing of the Source Code and the output. These subjects are cumulative meaning one builds for the next.

Competency: 3. **Write code in the COBOL programming language.**

Competency: 4. **Use INPUT/OUTPUT commands in a COBOL program when and where appropriate.**

Competency: 5. **Use ARITHMETIC OPERATIONS in a COBOL program when and where appropriate.**

Competency: 6. **Use COMPARING commands in a COBOL program including nested versions when and where appropriate.**

Competency: 7. **Write COBOL code to perform FINAL TOTALS.**

Competency: 8. **Apply REPORT and DATA EDITING when and where appropriate.**

Competency: 9. **Write COBOL code to perform CONTROL BREAKS including multiple level.**

Competency: 10. **Write COBOL code for TABLE PROCESSING including external tables.**

Competency: 11. **Write COBOL code to enable TABLE SEARCHING.**

Competency: 12. **Write COBOL code to enable SORTING.**

Learning Activities:

_____1. Read the assigned chapter for the subject to be implemented.

_____2. Participate in the class lecture/demo.

_____3. Desk-run the sample program.

_____4. Develop algorithm for assigned program.

_____5. Code assigned program, using algorithm.

_____6. Execute assigned program.

_____7. Debug assigned program.

_____8. Run assigned program successfully.

_____9. Print source code and resulting output, to turn in for credit.

Performance Assessment Activities:

_____1. Prepare program for required assignment.

_____2. Run program successfully.

_____3. Turn-in printed source code and output.

_____4. Complete successful programs to desired grade level.

_____5. Complete quizes and final exam successfully.