

LAST REVIEW:

2009-2010

NEXT REVIEW: 2014-15

2014-2015

STATUS:

A

COURSE TITLE: Introduction to Computer Programming

COMMON COURSE NUMBER: COP 1000C

CREDIT HOURS: 3

CONTACT HOUR BREAKDOWN

(per 16 week term)

CLOCK HOURS:

(Voc. Course ONLY)

Lecture: **48**

Lab: **16**

Clinic:

Other:

PREREQUISITE(S): MAT 0024

COREQUISITE(S): None

PRE/COREQUISITE(S):

COURSE DESCRIPTION *(750 characters, maximum):*

This course provides the beginning programming student with the techniques necessary to write well-documented, structured computer programs. The course is intended to emphasize the planning process using examples involving sequence, selection, and iteration. The course is designed to promote good programming practices for further study of programming languages. Upon successful completion of this course, the students should be able to design simple, structure, well-documented computer programs.

General Education Requirements – Associate of Arts Degree (AA), meets Area(s): Area

General Education Requirements – Associate in Science Degree (AS), meets Area(s): Area

General Education Requirements – Associate in Applied Science Degree (AAS), meets Area(s): Area

UNIT TITLES

1. Program Design
2. Pseudo-code, Flowcharts, and Other Graphical Representations
3. Developing an Algorithm
4. Selection Control Structures
5. Repetition Control Structures
6. Algorithms Using Sequence, Selection, and Repetition
7. Modularization
8. Communication Between Modules
9. Cohesion and Coupling
10. Primitive Data Structures

EVALUATION:

Please provide a brief description (250 characters maximum) that details how students will be evaluated on the course outcomes.

Evaluation instruments will include written and/or skills-based examinations and individual in-class and/or take-home assignments. Evaluation methods may also include group in-class and/or take-home assignments.

Common Course Number: COP 1000C

UNITS

Unit 1 Program Design

General Outcome:

1.0 The student shall:

Be able to describe the steps in the program development process, explain structured programming, and introduce algorithms using pseudo-code, flowcharts, or other graphical representations.

Specific Measurable Learning Outcomes:

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

- 1.1** Define the steps in program development.
- 1.2** Define structured programming.
- 1.3** Explain simple algorithms written in pseudo-code, flowcharts, or other graphical representations.

Common Course Number: COP 1000C

Unit 2 Pseudo-code, Flowcharts, and Other Graphical Representations

General Outcome:

2.0 The student shall:

Be able to demonstrate an understanding of the common words and keywords used when writing pseudo-code; the Structure Theorem and the three basic control structures.

Specific Measurable Learning Outcomes:

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

- 2.1** Write algorithms using pseudo-code, flowcharts, or other graphical representations.
- 2.2** Define the Structure Theorem

Common Course Number: COP 1000C

Unit 3 Developing an Algorithm

General Outcome:

3.0 The student shall:

Be able to develop an understanding of the methods of analyzing a problem and developing a solution.

Specific Measurable Learning Outcomes:

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

- 3.1** Define the problem.
- 3.2** Design a solution algorithm using the sequence control structure.
- 3.3** Check the solution algorithm.

Common Course Number: COP 1000C

Unit 4 Selection Control Structures

General Outcome:

4.0 The student shall:

Be able to explain the selection control structure by introducing simple and multiple selection, nested selection, and case construct.

Specific Measurable Learning Outcomes:

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

- 4.1** Define the selection control structures
- 4.2** Write algorithms using selection
- 4.3** Use the case structure

Common Course Number: COP 1000C

Unit 5 Repetition Control Structures

General Outcome:

5.0 The student shall:

Be able to develop algorithms which use the repetition control structure in the form of DOWHILE, REPEAT...UNTIL, and counted repetition loops.

Specific Measurable Learning Outcomes:

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

- 5.1** Employ repetition in simple programs using the DOWHILE construct.
- 5.2** Employ repetition in simple programs using the REPEAT...UNTIL construct.
- 5.3** Employ count-controlled repetition constructs in simple programs.

Common Course Number: COP 1000C

Unit 6 Algorithms Using Sequence, Selection, and Repetition

General Outcome:

6.0 The student shall:

Be able to develop algorithms to several programming problems using combinations of sequence, selection, and repetition constructs.

Specific Measurable Learning Outcomes:

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

- 6.1** Define complex problems.
- 6.2** Develop algorithms to solve complex problems.

Common Course Number: COP 1000C

Unit 7 Modularization

General Outcome:

7.0 The student shall:

Be able to develop an understanding of modularization as a means of dividing a problem into subtasks.

Specific Measurable Learning Outcomes:

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

- 7.1 Define modularization.
- 7.2 Create hierarchy charts or structure charts.
- 7.3 Define the steps in modularization.

Common Course Number: COP 1000C

Unit 8 Communication Between Modules

General Outcome:

8.0 The student shall:

Be able to define elementary data items, data structures, the concepts of inter-module communication, local and global data, and the passing of parameters between modules.

Specific Measurable Learning Outcomes:

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

- 8.1** Use program data.
- 8.2** Define communication between modules.
- 8.3** Use parameters in program design.
- 8.4** Define object-oriented design.

Common Course Number: COP 1000C

Unit 9 Cohesion and Coupling

General Outcome:

9.0 The student shall:

Be able to develop an understanding of the concepts of module cohesion and coupling.

Specific Measurable Learning Outcomes:

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

- 9.1** Define module cohesion.
- 9.2** Define module coupling.

Common Course Number: COP 1000C

Unit 10 Primitive Data Structures

General Outcome:

10.0 The student shall:

Be able to develop an understanding of the concepts of arrays, multi-dimensional arrays, records, arrays of records, and the algorithms that manipulate these data structures.

Specific Measurable Learning Outcomes:

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

- 10.1** Define an array
- 10.2** Define a multi-dimensional array
- 10.3** Define a record
- 10.4** Define an array of records