

COMS-120: INTRODUCTION TO PROGRAMMING CONCEPTS AND METHODOLOGIES

Effective Term

Fall 2021

BOT Approval

03/11/2021

SECTION A - Course Data Elements**CB04 Credit Status**

Credit - Degree Applicable

Discipline

Minimum Qualifications	And/Or
Computer Science (Master's Degree)	

Subject Code

COMS - Computer Studies

Course Number

120

Department

Computer Studies

Division

Business and Computer Studies

Full Course Title

Introduction to Programming Concepts And Methodologies

Short Title

Intro Prog Concepts & Method

CB03 TOP Code

0707.10 - *Computer Programming

CB08 Basic Skills Status

NBS - Not Basic Skills

CB09 SAM Code

C - Clearly Occupational

Rationale

Removed Recommended Preparation MATH 90 per AB

SECTION B - Course Description**Catalog Course Description**

This course is an introduction to the basic principles of programming using an object-oriented programming language. Topics include the software development life-cycle, program design tools and programming environments, documentation, coding, data types, arrays, control structure, algorithms, file input/output, error handling, parameters, as well as principles of testing and designing test data.

SECTION C - Conditions on Enrollment**Open Entry/Open Exit**

No

Repeatability

Not Repeatable

Grading Options

Letter Grade or Pass/No Pass

Allow Audit

Yes

Requisites**SECTION D - Course Standards****Is this course variable unit?**

No

Units

3.00000

Lecture Hours

54.00

Outside of Class Hours

108

Total Contact Hours

54

Total Student Hours

162

Distance Education Approval**Is this course offered through Distance Education?**

Yes

Online Delivery Methods

DE Modalities	Permanent or Emergency Only?
Entirely Online	Permanent
Hybrid	Permanent
Online with Proctored Exams	Permanent

SECTION E - Course Content**Student Learning Outcomes****Upon satisfactory completion of the course, students will be able to:**

1. Define the software development life cycle and principles of structured programming then design, code, test, and debug a program.

Course Objectives**Upon satisfactory completion of the course, students will be able to:**

1. Describe the software development life cycle.
2. Describe the principles of structured programming and be able to describe, design, implement, and test structured programs using currently accepted methodology.
3. Explain what an algorithm is and its importance in computer programming.

Course Content

1. Software life cycle including design, development, styles, documentation, testing and maintenance.
2. Procedural versus objected-oriented programming
 - a. Survey of current languages

3. Program design tools and programming environments
4. Documentation
5. Coding conventions
6. Data types, variables, expressions, and sequential processing
7. Arrays
 - a. Declaring and allocating arrays
 - b. Multiple-subscripted arrays
8. Control structure
 - a. Selective structures: if and switch
 - b. Repetitive structures: loops
9. Algorithms including simple sorting and searching.
 - a. File I/O Files and streams
 - b. Sequential access files
10. Error handling
11. Passing parameters by value and by reference
12. Principles of testing and designing test data

Methods of Instruction

Methods of Instruction

Types	Examples of learning activities
Lecture	Example: Lecture on the history of BASIC programming.

Online Adaptation

Types	Examples of learning activities
Activity	
Directed Study	
Discussion	
Group Work	
Individualized Instruction	
Journal	
Lecture	

Instructor-Initiated Online Contact Types

Announcements/Bulletin Boards
 Chat Rooms
 Discussion Boards
 E-mail Communication
 Telephone Conversations
 Video or Teleconferencing

Student-Initiated Online Contact Types

Chat Rooms
 Discussions
 Group Work

Course design is accessible

Yes

Methods of Evaluation

Methods of Evaluation

Types	Examples of classroom assessments
Exams/Tests	Complete 4 to 6 exams.
Projects	Create a program to compute inflation.
Other	Class Work -- Complete Tutorial 1.
Homework	Answer selected Review Questions and Programming Questions at the end of the chapter. (See chapter 3 in book.)

Other

Final Exam -- Complete a 50-question multiple choice and true/false based exam.
Mid Term -- Complete a 50-question multiple choice and true/false based exam.

Assignments

Reading Assignments

Read assigned chapters from the textbook.

Examples using the Deitel book include:

Read Chapter 1 Introduction to Computers and C++.

Read Chapter 2 Introduction to C++ Programming: Input/Output and Operators.

Writing Assignments

Write assignments from the textbook. Complete assigned lab exercises and projects.

1. Example of Lab Assignment:

Code a Card Shuffling and Dealing Simulation Program. (See Page 882 in the Deitel book)

2. Example of Homework assignment:

Complete the following Case Study: A Transaction-Processing Program. (See Page 621 in the Deitel book)

SECTION F - Textbooks and Instructional Materials

Material Type

Textbook

Author

Deitel, P.

Title

C++ How to Program

Edition/Version

10th

Publisher

Pearson

Year

2016

SECTION G - Diversity, Equity and Inclusivity

Course Codes (Admin Only)

CB00 State ID

CCC000169494

CB10 Cooperative Work Experience Status

N - Is Not Part of a Cooperative Work Experience Education Program

CB11 Course Classification Status

Y - Credit Course

CB13 Special Class Status

N - The Course is Not an Approved Special Class

CB23 Funding Agency Category

Y - Not Applicable (Funding Not Used)

CB24 Program Course Status

Program Applicable

Allow Pass/No Pass

Yes

Only Pass/No Pass

No