COM120 Programming Concepts and Logic

4 UNITS: 30 HOURS LECTURE + 20 HOURS LAB

SYLLABUS

Description

This course introduces the fundamentals of programming logic, program flow and the control statements needed to implement a programming solution and write an algorithm. The course covers problem analysis and definition, program design, flowcharting, validation techniques, testing techniques, and the basic features of computer hardware, software, and data.

Prerequisites

None

Resources

Note: Be aware some websites may change with no notice

Text

Robertson, L. (2004) Simple Program Design, (4th ed.). Cambridge: Course Technology.

Required Resources

None

Supplemental Resources

Farrell, J. (2004). Programming Logic and Design, Comprehensive, (4th ed.). Cambridge: Course Technology.

Course Objectives and Outcomes

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

Produce

- Logical Algorithms as solutions to business problems
- Modularized, structured, and automated solutions to business problems
- Appropriate business documentation using CASE tools (IPO chart, pseudocode, flowcharts)

Use

- The three basic control structures: sequence, selection, and iteration (repetition)
- Arrays, functions, data structures, pointers, and file I/O
- Project development and documentation tools (IPO charts, pseudocode, flowcharts, data flow diagrams, top down development, modularization and structured design concepts)
- Binary, decimal and hexadecimal number systems
- Oral and written instructions to determine requirements, procedures, expected outcomes and products
- Pseudocode algorithms to solve basic business computing problems
- Effective team techniques to work with a diverse group of individuals toward a common goal
- Commonly used office productivity software

Knowledgeably Discuss

- Basic computer concepts
- Fundamentals of programming
- Designing simple programs demonstrating good programming style
- Program control statements
- Internal data representation
- Software solution development and integration of information technology into the business model
- The use of CASE tools to document the development of business software solutions
- Basic control structures, passing arguments, binary, decimal and hexadecimal numbering systems and common algorithms

Course Outline (see course schedule for specific dates)

Week			
	Topic	**Reading**	**Project Assigned**
1	Day 1: Introduction to course Introduction to Computers	Chapter 1-3 in Robertson	Read: Robertson CH 1-3 30 pages, 3 hours
	Lesson 1: Data Representations (SM)	Student Manual: Lesson 1 Robertson: Appendix 1 239-245	**Project 1:** Number System Conversions and binary math: 40 parts, 4 hours **Evaluation:** Graded , 4 points **Read:** SM Lesson 1 9 pages, 0.9 hours **Knowledge Check:** Lesson 1 20 Questions, 1.3 hours **Evaluation:** Graded, 1point
2	Algorithm Design, Pseudocode and Flow Charts	Student Manual: Lesson 2 Robertson: Appendix 1, 239-245	**Project 2 :** Manipulate 3 Numbers, 1 hour **Evaluation:** Graded, 4 points **Project 3:** Weighted Grade Calculation, 1 hour **Evaluation:** Graded, 4 points **Read:** SM Lesson 2 6 pages, 0.6 hours Robertson: Appendix 1 7 pages, 0.7 hours **Knowledge Check:** Lesson 2 20 Questions, 1.3 hours **Evaluation:** Graded, 1 point
3	Lesson 3: Expressions and Intro. to Selection	SM: Lesson 3 Robertson: Chapter 4 Roberson: Appendix 1, 246-253	**Project 4:** Evaluating Boolean Expressions, 2 hours **Evaluation:** Graded, 4 points **Project 5:** Calculate Weekly Pay, 1 hour **Evaluation:** Graded, 4 points **Read:** SM: Lesson 3 6 pages, 0.6 hours Robertson: Chapter 4 16 pages, 1.6 hours Robertson: Appendix 1, 246-253 8 pages, 0.8 hours **Knowledge Check:** Lesson 3

			18 Questions, 1.2 hours **Evaluation:** Graded, 1 point
4	Lesson 4: Nested Selection Algorithms	SM: Lesson 4 Robertson: Appendix 1, 254 - 255	**Project 6:** Calculating Age, 1 hour **Evaluation:** Graded, 4 points **Project 7:** Valid Date, 2 hours **Evaluation:** Graded, 4 points **Read:** SM : Lesson 4 6 pages, 0.6 hours Robertson: Appendix 1, 254-255 2 pages, 0.2 hours **Knowledge Check:** Lesson 4 15 Questions, 1 hour **Evaluation:** Graded, 1 point
5	Lesson 5: Introduction to Iteration Midterm: Lessons 1-4 20 points	SM: Lesson 5 Robertson: Chapter 5	**Project 8:** Number Progression, 1 hour **Evaluation**: Graded, 4 points **Read:** Student Manual Lesson 5 9 pages, 0.9 hours Robertson: Chapter 5 16 pages, 1.6 hours **Knowledge Check:** Lesson 5 15 Questions, 1.5 hours **Evaluation:** Graded, 1 point **Midterm** 20 Questions **Evaluation:** Graded, 20 points
6	Lesson 6: Input Validation	SM: Lesson 6	**Project 9:** Investment, 2 hours Evaluation: Graded, 4 points **Read:** SM Lesson 6 9 pages, 0.9 hours
7	Lesson 7: Modularization	SM: Lesson 7 Robertson: Chapters 8 and 9	**Project 10**: Payroll Framework 5 hours **Evaluation**: Graded, 4 points **Read:** Student Manual: Lesson 7 9 pages, 0.9 hours Robertson: Chapters 8 and 9 45 pages, 4.5 hours **Knowledge Check:** Lesson 7 16 Questions, 1 hour Evaluation: Graded, 1 point
	Lesson 8: Object Oriented Design	SM: Lesson 8 Roberson: Chapter 11	**Project 11:** Class Design 4 hours **Evaluation:** Graded, 4 points **Read:** Student Manual: Lesson

8			8 12 pages, 1.2 hours Robertion: Chapter 11 22 pages, 2.2 hours **Knowledge Check:** Lesson 8 20 Questions, 1.3 hours **Evaluation:** Graded, 1 points
9	Lesson 9: Arrays	Student Manual: Lesson 9 Robertson: Chapter 7	**Project 12:** Manipulate Arrays 1 hour **Evaluation:** Graded, 4 points **Project 13**: Payroll Full Pseudocode 4 hours Evaluation: Graded, 4points **Read:** Student Manual Lesson 9 9 pages, 0.9 hours Robertson: Chapter 7 18 pages, 1.8 hours **Knowledge Check:** Lesson 9 15 Questions, 1 hour
10	Perspectives Final, 20 points		**Final** 20 Questions **Evaluation:** Graded, 20 points 20 points

Evaluation

Knowledge Checks	8%
Projects	52%
Tests	40%
Total	**100%**

Coleman University Grading Scale

Letter Grade	Percentage	Grade Points
A	94-100	4.0
A-	90-93	3.7
B+	87-89	3.3
B	84-86	3.0
B-	80-83	2.7
C+	77-79	2.3
C	74-76	2.0
C-	70-73	1.7
F	0-69	0
INC		0
W		0
WP		0
WF		0
CR	74 or above	0
NC	73 or below	0
PASS	70 or above	0

Instruction Methods

☒ Lecture	☒ Labs	☐ Student Presentations
☒ Reading	☒ Projects	☒ Team Environment
☐ Exercises	☒ Online Complements	
☐ Research	☒ Written Assignments	

Your Grades for this Course

Your final grade for this course will be based on an assessment by the Instructor of your performance on a number of course activities, which may include objective tests, classroom exercises, laboratory demonstrations, project papers, or other types of activities. The chart below indicates in what activities you will engage, how many possible points can be earned for each activity, and the percentage of your final grade that will be accounted for by each activity.

Students in this course should be graded following Coleman University assessment practices and policies. A point system is used in the University to indicate student performance on various required activities or projects. For this course, it is recommended that points be distributed as follows:

Coleman University Grade Assignment Policy:

Percent	Letter Grade	Grade Points
94-100	A	4
90-93	A-	3.67
87-89	B+	3.33
84-86	B	3
80-83	B-	2.67
77-79	C+	2.33
74-76	C	2
70-73	C-	1.67
67-69	D+	1.33
64-66	D	1
60-63	D-	0.67
N/A	INC	0
N/A	W	0
60 or above	CR	0
59 or below	NC	0
N/A	I	0
N/A	W	0
N/A	AU	0
N/A	TR	0
N/A	WV	0

Legend	
CR = Credit	NC = No Credit
I = Incomplete	W = Course Withdrawal

AU = Audit	TR = Transfer Credit
WV = Waiver	

Academic Accommodation / Adjustment Policy:

In accordance with Section 504 of the Rehabilitation Act of 1973 and the Americans with Disabilities Act (ADA), Coleman University offers accommodations to students with documented physical, psychological, and/or cognitive disabilities. Coleman University will adhere to all applicable federal, state, and local laws, regulations, and guidelines with respect to providing reasonable accommodations as required to offer equal educational opportunities to qualified disabled individuals.

To qualify for an academic accommodation under ADA, the student must provide adequate documentation of a disability. Students seeking academic accommodations should contact the campus ADA Coordinator at 858-966-3953 or via email at ada@coleman.edu. The ADA Coordinator will review the documentation provided and verify ADA coverage. Students covered under ADA must meet with the ADA Coordinator at the beginning of every term to determine the appropriate academic accommodations. Failing to meet with the ADA Coordinator at the beginning of every term may impact the availability of accommodations.

After the academic accommodations have been determined, the students' instructors will be notified by the ADA Coordinator. If any problems or concerns regarding the provision of accommodations occur, the student must inform the ADA Coordinator. If the student feels accommodation is not being made appropriately, the student may follow the published Student Grievance Procedures.