Course Specification
(Bachelor)

Course Title: Introduction to Programming	
Course Code: CSC 1103	
Program Bachelor in Computer Science	
Department: Department of Computer Science	
College: Computers and Information Technology	
Institution: University of Tabuk	
Version: 1.0	
Last Revision Date: 27 July 2022	

Table of Contents

A. General information about the course:

1. Course Identification

1. Credit hours: (3)

2. Course type

A.	☐ University	☒ College	☐ Department	☐ Track	☐ Others
B.	☒ Required		☐ Elective		

3. Level/year at which this course is offered: (Second Semester / First Year)

4. Course general Description:

This course introduces the students to the main concepts and techniques used for developing simple programs. The course covers and discusses through examples variables and data types, input and output statements, expressions, control statements, functions and sub-programs, data types collections. Through the course, the students will learn how to write, design, test, debug and document programs in an integrated program development.

5. Pre-requirements for this course (if any):

CSC 1102 Problem Solving in Computing

6. Co-requisites for this course (if any):

NA

7. Course Main Objective(s):

After completing this course, the students will be able to:
1) Understand the fundamental concepts of programming.
2) Use Integrated Development Environment (IDE) to write, compiles, test, and debug programs.
3) Develop programs for problem solving at a basic and moderate level.
4) Understand differences among available control statements for implementing a solution
5) Analyze problems to design reasonable and efficient programs

2. Teaching mode (mark all that apply)

No	Mode of Instruction	Contact Hours	Percentage
1	Traditional classroom	60	100%
2	E-learning		
3	Hybrid • Traditional classroom • E-learning		

No	Mode of Instruction	Contact Hours	Percentage
4	Distance learning		

3. Contact Hours (based on the academic semester)

No	Activity	Contact Hours
1.	**Lectures**	30
2.	**Laboratory/Studio**	30
3.	**Field**	
4.	**Tutorial**	
5.	**Others (specify)**	
Total		60

B. Course Learning Outcomes (CLOs), Teaching Strategies and Assessment Methods

Code	Course Learning Outcomes	Code of PLOs aligned with program	Teaching Strategies	Assessment Methods
1.0	**Knowledge and understanding**			
1.1	Recognize techniques used to write programs and to design solutions	ILO1, ILO 2	Lectures Class Discussions	Exams
1.2	know how programming can be used to solve problems and how problem can be represented in a way consumable by programs	ILO1, ILO 2	Lectures Class Discussions	Exams
1.3	Describe different control structures used in writing computer programs	ILO1, ILO 2	Lectures Class Discussions	Exams
1.4	Identify how problem is decomposed in terms of data types, control structure, modularization and data types collections	ILO1, ILO 2	Lectures Class Discussions	Exams
2.0	**Skills**			

Code	Course Learning Outcomes	Code of PLOs aligned with program	Teaching Strategies	Assessment Methods
2.1	Develop thoughts about basic programming skills in implementing solutions using suitable data types and constructs	ILO 2	Lectures Class Discussions Labs	Exams Lab. assignments
2.2	Assess code written by others	ILO 2	Lectures Class Discussions Labs	Exams Lab. assignments
2.3	Apply appropriate design, coding, testing, and documenting of computer programs under a specific IDE	ILO 2	Lectures Class Discussions Labs	Exams Lab. assignments
2.4	Apply appropriate logic and computer syntax for making solution efficient, correct, concise and clear	ILO 2	Lectures Class Discussions Labs	Exams Lab. assignments
3.0	Values, autonomy, and responsibility			
3.1	Use communication and teamwork skills	ILO9, ILO10	Class Discussions Labs	Lab. assignments
3.2	Practice computer ethics effectively and professionally	ILO2	Class Discussions Labs	Lab. assignments

C. Course Content

No	List of Topics	Contact Hours
1.	**Elementary Programming (Part 1): Identifiers, Variables, Assignment Statements and Assignment Expressions, Named Constants, Naming Conventions, Numeric Data Types and Operations, Numeric Literals Lab: Introducing the IDE to the students and how to compile programs**	**4**
2.	**Elementary Programming (Part 2): Evaluating Expressions and Operator Precedence, Increment and Decrement Operators, Numeric Type Conversions, Reading Input from the Console, Writing Output to Standard Console Lab: Writing simple programs that contain variables and constants declaration, using input, output and assignment statements**	**4**
3	**Selection (Part 1): Boolean Data Type, if Statements, Two-Way if-else**	**4**

	Statements, Nested if and Multi-Way if-else Statements Lab: Implementing different forms of selection(if else – case – nested ifs)	
4	Selection (Part 2): Generating Random Numbers, Logical Operators, switch Statements, Conditional Expressions. Lab: Implementing Generating Random Numbers, Logical Operators and switch Statements	4
5	Mathematical Functions, Characters, and Strings: Common Mathematical Functions Lab: Implementing the above topic	4
6	Character Data Type and Operations, The String Type, Formatting Console Output Lab: Implementing how to handle character and; string data types and how to format output	4
7	Loops (Part 1): Introduction, The while Loop, The do-while Loop Lab: Implementing while and do while loops with different examples	4
8	Loops (Part 2): The for Loop, Which Loop to Use Lab: Implementing for loop and which loop to use	4
9	Loops (Part 3): Nested Loops, Keywords break and continue Lab: Implementing the above topic	4
10	Methods (Part 1): Introduction, Defining a Method, Calling a Method , void Method Example Lab: Implementing simple methods and calling them into simple programs	4
11	Methods (Part 2): Passing Arguments by Values, Modularizing Code, Overloading Methods, The Scope of Variables Lab: Implementing the above topics	4
12	Methods (Part 3): Recursion Lab: Implementing the recursion	4
13	Single-Dimensional Arrays: Introduction, Array Basics, Copying Arrays, Passing Arrays to Methods, Returning an Array from a Method Lab: Implementing single - Dimensional Arrays and their use with the methods	4
14	Two-Dimensional Array Basics, Processing Two-Dimensional Arrays, Passing Two-Dimensional Arrays to Methods Lab: Implementing Two - Dimensional Arrays and their use with the methods	4
15	Data Types Collections: String and String Tokenizers, ArrayLists, Dictionary and HashTable Lab: Implementing the mentioned data types collections	4
Total		**60**

D. Students Assessment Activities

No	Assessment Activities *	Assessment timing (in week no)	Percentage of Total Assessment Score
1.	Lab Assignment	2- 15	20%
2.	Lab Exam	14 or 15	10%
3.	Mid-Exam 1	6 or 7	15%
4.	Mid-Exam 2	11 or 12	15%
5.	Final Exam	16 or 17	40%

*Assessment Activities (i.e., Written test, oral test, oral presentation, group project, essay, etc.).

E. Learning Resources and Facilities

1. References and Learning Resources

Essential References	Y. Daniel Liang, 2019, Intro to Java Programming, Comprehensive Version, Student Value Edition 12th Edition. Pearson, ISBN-10 : 0136520154 ISBN-13: 978-0136520153.
Supportive References	Harvey M. Deitel and Paul J. Deitel, C++ How to Program, , 2016, 10th edition, ASIN: 0134448235 PROGRAMMING LANGUAGES ACADEMY, Python for Beginners: 2 Books in 1: Python Programming for Beginners, Independently published, 2020, ISBN-10 : 1654414018, ISBN-13 : 978-1654414016
Electronic Materials	
Other Learning Materials	

2. Required Facilities and equipment

Items	Resources
facilities (Classrooms, laboratories, exhibition rooms, simulation rooms, etc.)	Fully equipped computer Lab
Technology equipment (projector, smart board, software)	- Projector with HDMI port - Operating System: Windows - Software: Java, C++, Python
Other equipment (depending on the nature of the specialty)	

F. Assessment of Course Quality

Assessment Areas/Issues	Assessor	Assessment Methods
Effectiveness of Teaching	Faculty, Program Leaders, and Advisory Board	Both Direct and Indirect
	Students	Indirect
Effectiveness of Students Assessment	Faculty, Program Leaders, Advisory Board, and Independent Opinion	Both Direct and Indirect
Quality of Learning Resources	Faculty, Students, and Advisory Board	Indirect
The Extent to which CLOs have been Achieved	Faculty, Program Leaders, Advisory Board, and Independent Opinion	Direct (as in section B) and Indirect/Surveys
	Students	Indirect
Other	-	-

Assessors (Students, Faculty, Program Leaders, Peer Reviewer, Others (specify)

Assessment Methods (Direct, Indirect)

G. Specification Approval

COUNCIL /COMMITTEE	
REFERENCE NO.	
DATE	