اعتماد
NCAAA

T4

2020

Course Specifications

Course Title:	Fundamentals of Programming
Course Code:	211CCS-4
Program:	BSc in Computer Science
Department:	Computer Science
College:	Computer Science and Information System
Institution:	Najran University

Table of Contents

A. Course Identification

1. Credit hours: **4** (3, 2, 1) [Theory, Lab, Tutorial]	

2. Course type

a.   University ☐   College ☑   Department ☐   Others ☐

b.   Required ☑   Elective ☐

3. Level/year at which this course is offered: Year 2 / Level 4

4. Pre-requisites for this course (if any)**:**

None

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

None

6. Mode of Instruction (mark all that apply)

No	Mode of Instruction	Contact Hours	Percentage
1	**Traditional classroom**	60	100%
2	**Blended**		
3	**E-learning**		
4	**Distance learning**		
5	**Other**		

7. Contact Hours (based on academic semester)

No	Activity	Contact Hours
1	**Lecture**	30
2	**Laboratory/Studio**	20
3	**Tutorial**	10
4	**Others** (specify)	
	Total	**60**

B. Course Objectives and Learning Outcomes

1. Course Description

This course offers an introduction to computer science and computer programming in Java. With emphasis on object-oriented programming (OOP) practice and problem-solving skills. The course presents a balanced coverage of java language basics, programming concepts and techniques and standard algorithms. The topics included are Algorithms, Flowcharts, Data types, basic java syntax, introduction to objects and classes, variables, Operators, Selection and control structures, Arrays, Loops.

2. Course Main Objective

Prepare the students with the basic concepts of programming so that he could read and understand any programming language and develop the required skills to build/write a program whenever they required.

3. Course Learning Outcomes

	CLOs	Aligned PLOs
1	**Knowledge and Understanding**	
1.1	Describe the basic concepts of programming	K_1
1.2		
1.3		
1...		
2	**Skills :**	
2.1	Construct programs with basic programming elements	S_1, S_2, S_4
2.2	Apply the concept of flowchart and algorithm in solving problems.	S_2, S_4, S_5
2.3	Apply function concepts of programs.	S_5
2.4	Create programs with advanced programming elements.	S_2, S_4, S_5
2.5	Assess program execution	S_2
3	**Values:**	
3.1		
3.2		
3.3		
3...		

C. Course Content

No	List of Topics	Contact Hours
1	Introduction to Programming language and its types, Introduction to assembler, interpreter and compiler	9
2	Introduction, Flowcharts, Algorithm	6
3	Elementary Programming	6
4	Mathematical Functions Characters and Strings	9
5	Selections	6
6	Loops	6
7	Methods	6
8	Arrays	6
9	Multi-Dimensional Arrays	6
Total		**60**

D. Teaching and Assessment

1. Alignment of Course Learning Outcomes with Teaching Strategies and Assessment Methods

Code	Course Learning Outcomes	Teaching Strategies	Assessment Methods
1.0	**Knowledge and Understanding**		
1.1	Describe the basic concepts of programming	TS:1-Interactive lectures using PowerPoint slides with more	• Locally Developed Exams such as Quiz, Mid

Code	Course Learning Outcomes	Teaching Strategies	Assessment Methods
		examples in the class TS:2- Engaging the students in problem-based learning through tutorials TS:3- Lab Demonstrations TS: 4 – Recall the topics discussed in the last lecture by asking questions to the students. TS: 5 – Associating the topics in each chapter with the CLO. TS:6 – Conducting oral quizzes by the end of each chapter TS:7 – Giving more example programs in the lecture TS: 8 – Discussion with the students in the class hours	& Final Exams with scoring rubrics • Assignments involving critical and logical thinking questions • Quizzes
1.2			
…			
2.0	**Skills**		
2.1	Construct programs with basic programming elements	TS:1-Interactive Lectures using PowerPoint slides with more examples TS:2- Engaging the students in problem-based learning through Tutorials TS:3- Lab Demonstrations TS: 4 – Recall the topics discussed in	• Locally Developed Exams such as Quiz, Mid & Final lab/theory Exams with scoring rubrics. • Assignments involving critical and logical

Code	Course Learning Outcomes	Teaching Strategies	Assessment Methods
		the last lecture by asking questions to the students. TS: 5 – Associating the topics in each chapter with the CLO. TS:6 – Conducting oral quizzes by the end of each chapter TS:7 – Giving more example programs in the lecture TS: 8 – Discussion with the students in the class hours	thinking programs • Giving lab exercises at the end of each lab activity
2.2	Apply the concept of flowchart and algorithm in solving problems.	TS:1-Interactive Lectures using PowerPoint slides with more examples TS:2- Engaging the students in problem-based learning through Tutorials TS: 3 – Discussion with the students in the class hours.	Locally Developed Exams such as Quiz, Mid Exam, Final Exam
2.3	Apply function concepts of programs.	TS:1-Interactive Lectures using PowerPoint slides with more examples in the class TS:2- Engaging the students in problem-based learning through Tutorials TS:3- Lab Demonstrations TS: 4 – Recall the topics discussed in the last lecture by asking questions	Quiz, Lab Assessment, Lab Final Exam, Final Exam.

Code	Course Learning Outcomes	Teaching Strategies	Assessment Methods
		to the students. TS: 5 – Providing online links (internet resources) in the slides at the end of each chapter to the students to know more about the topics discussed in the lecture TS:6– Conducting oral quizzes by dividing the students into groups and asking them the questions at the end of the lecture involving complex topics programs in the lecture and asking the students to write a program at the end of lecture TS: 8 – Discussion with the students in the class hours.	

Code	Course Learning Outcomes	Teaching Strategies	Assessment Methods
2.4	Create programs with advanced programming elements.	TS:1-Interactive Lectures using PowerPoint slides with more examples TS:2- Engaging the students in problem-based learning through Tutorials TS:3- Lab Demonstrations TS: 4 – Recall the topics discussed in the last lecture by asking questions to the students. TS: 5 – Providing online links (internet resources) in the slides at the end of each chapter to the students to know more about the topics discussed in the lecture TS:6 – Conducting oral quizzes by dividing the students into groups and asking them the questions at the end of the lecture involving complex topics TS:7 – Giving more example programs in the lecture and asking the students to write a program at the end of lecture TS: 8 – Discussion with the students in the class hours.	• Locally Developed Exams such as Quiz, Mid & Final lab/theory Exams with scoring rubrics • Assignments involving critical and logical thinking programs • Giving lab exercises at the end of each lab activity
2.5	Assess program execution		
3.0	**Values**		
3.1			
3.2			
…			

2. Assessment Tasks for Students

#	Assessment task*	Week Due	Percentage of Total Assessment Score
1	Quizzes	3rd and 9th week	10%
2	Midterm Exam	6th week	20%
3	Lab Activities	6th week	10%
4	Assignments or mini project (presentation)	8th week	10%
5	Final Lab Exam	11th week	10%
6	Final Theory Exam	12th and 13th week	40%

*Assessment task (i.e., written test, oral test, oral presentation, group project, essay, etc.)

E. Student Academic Counseling and Support

Arrangements for availability of faculty and teaching staff for individual student consultations and academic advice :

- 10 weekly office hours + appointments
- 4 weekly academic advising hours
- Extra weekly 2 office hours prior to exams.

F. Learning Resources and Facilities

1.Learning Resources

Required Textbooks	Liang, Y. Daniel. *Introduction to Java Programming and Data Structures, Comprehensive Version, 12th edition*. Pearson Education, 2020.
Essential References Materials	
Electronic Materials	
Other Learning Materials	NetBeans with JDK 8 as an IDE

2. Facilities Required

Item	Resources
Accommodation (Classrooms, laboratories, demonstration rooms/labs, etc.)	Lecture Room with +30 seats with PC, Auto Projector and a white board
Technology Resources (AV, data show, Smart Board, software, etc.)	• The laboratory equipped with network, so that the students have their privacy in accessing their own files with limited permissions of accessibility. • Projectors installed in the labs to carry out lab demonstrations and presentations

Item	Resources
Other Resources (Specify, e.g. if specific laboratory equipment is required, list requirements or attach a list)	

G. Course Quality Evaluation

Evaluation Areas/Issues	Evaluators	Evaluation Methods
1. Strategies for Obtaining Student Feedback on Effectiveness of Teaching	● Students course evaluation through CLO course survey during the course period. ● Organizing general meetings with students.	
2. Other Strategies for Evaluation of Teaching by the Instructor or by the Department	● Consulting peers on teaching. ● Discussion about the course in department.	
3. Processes for Improvement of Teaching	● Follow the recommendations provided by the curriculum committee at the end of the previous semester about the course. ● Encouraging the students to participate in active learning by offering tutorials and assignments.	
4. Describe the planning arrangements for periodically reviewing course effectiveness and planning for improvement.	● Preparing course report and action plans every semester. ● Planning to update course syllabus every two years (the course coordinator are reviewing the courses and the references). ● Curriculum committee updates the syllabus based on the student outcomes.	

Evaluation areas (e.g., Effectiveness of teaching and assessment, Extent of achievement of course learning outcomes, Quality of learning resources, etc.)
Evaluators (Students, Faculty, Program Leaders, Peer Reviewer, Others (specify)
Assessment Methods (Direct, Indirect)

H. Specification Approval Data

Council / Committee	Computer Science Departmental Council
Reference No.	14440203-0185-00002
Date	1st Sep, 2022