

QFXX/0408-4.0E	Course Plan for Bachelor program - Study Plan Development and Updating Procedures/ Computer Science Department
----------------	---

Study plan No.	2021-2020		University Specialization		Computer science	
Course No.	0112120		Course name		Programming Principles	
Credit Hours	3 Hours		Prerequisite Co-requisite		Introduction to Information Technology	
Course type	<input type="checkbox"/> MANDATORY UNIVERSITY REQUIREMENT	<input type="checkbox"/> UNIVERSITY ELECTIVE REQUIREMENTS	<input type="checkbox"/> FACULTY MANDATORY REQUIREMENT	<input type="checkbox"/> Support course family requirements	<input type="checkbox"/> Mandatory requirements	<input type="checkbox"/> Elective requirements
Teaching style	<input type="checkbox"/> Full online learning		<input type="checkbox"/> Blended learning		<input type="checkbox"/> Traditional learning	
Teaching model	<input type="checkbox"/> 2Synchronous: 1asynchronous		<input type="checkbox"/> 2 face to face : 1synchronous		<input type="checkbox"/> 3 Traditional	

Faculty member and study divisions information (to be filled in each semester by the subject instructor)

Name	Academic rank	Office No.	Phone No.	E-mail	
To be filled by the instructor					
Division number	Time	Place	Number of students	Teaching style	Approved model
To be filled by the instructor					

Brief description

This course gives an introduction to programming using JAVA in which the following concepts are presented: input/output operations, relational and logical operators, variables and constants, control statements (selection and repetition), methods and arrays.

Learning resources

Course book information (Title, author, date of issue, publisher ... etc)	1. Y. Daniel Liang: Introduction to Java Programming, Eleventh Edition, Global Version. 2017 2. D. S Malik: Java Programming from problem analysis to program design, 5 th Edition 2011.				
Supportive learning resources (Books, databases, periodicals, software, applications, others)	Java™ How to Program, 9th Edition, By H. M. Deitel. & P. J. Deitel , Prentice Hall, 2012.				
Supporting websites	https://www.w3schools.com/java/java_intro.asp				
The physical environment for teaching	<input type="checkbox"/> Class room	<input type="checkbox"/> labs	<input type="checkbox"/> Virtual educational platform	<input type="checkbox"/> Others	

QFXX/0408-4.0E	Course Plan for Bachelor program - Study Plan Development and Updating Procedures/ Computer Science Department
----------------	---

Necessary equipment and software	Net Beans 8.2 (https://www.oracle.com/technetwork/java/javase/downloads/jdk-netbeans-jsp-3413139-esa.html)
Supporting people with special needs	-----
For technical support	-----

Course learning outcomes (S= Skills, C= Competences K= Knowledge,)

No.	Course learning outcomes	The associated program learning output code
Knowledge		
K1	Analyze a given problem statements.	MK2
K2	Write clear, elementary java programs.	MK2
K3	Understand control structures.	MK2
K4	Explain built-in and programmer-defined Methods.	MK2
K5	Explain arrays	MK2
Skills		
S1	Introduction to computers, programs and why JAVA.	MS2
S2	Applying the fundamental programming aspects (input/output, types of variables, constants, equations, increment, decrement, assignment, relational, equality and logical operators)	MS2
S3	Design, develop, and document well-structured programs.	MS2
S4	Using java code of "if" statement, "switch case" statement and nested conditional statements.	MS2
S5	Writing java code of repetition structures (for loop, while loop) and nested loops.	MS2
S6	Using pre-defined methods.	MS2
S7	Writing user-defined methods (void and return method) and understanding parameters.	MS2
S8	Defining, creating and manipulating arrays.	
Competences		
C1	Ability to analyze a given problem statements.	MC3
C2	Ability to write clear, elementary java programs.	MC3
C3	Ability to write java programs using any control structures.	MC3
C4	Ability to write java programs using built-in and programmer-defined Methods.	MC3
C5	Ability to write java programs using arrays.	MC3

Mechanisms for direct evaluation of learning outcomes

Type of assessment / learning style	Fully electronic learning	Blended learning	Traditional Learning (Theory Learning)	Traditional Learning (Practical Learning)
First exam	0	0	%20	0
Second / midterm exam	%30	%30	%20	30%

QFXX/0408-4.0E	Course Plan for Bachelor program - Study Plan Development and Updating Procedures/ Computer Science Department			
Participation / practical applications	0	0	10	30%
Asynchronous interactive activities	%30	%30	0	0
final exam	%40	%40	%50	40%

Note: Asynchronous interactive activities are activities, tasks, projects, assignments, research, studies, projects, work within student groups ... etc, which the student carries out on his own, through the virtual platform without a direct encounter with the subject teacher.

Schedule of simultaneous / face-to-face encounters and their topics

Week	Subject	learning style*	Reference **
1	Introduction to Computers, Programs, and Java 1.1 Introduction 24 1.2 What Is a Computer? 24 1.3 Programming Languages 29 1.4 Operating Systems 31 1.5 Java, the World Wide Web, and Beyond 32 1.6 The Java Language Specification, API, JDK, JRE, and IDE 33 1.7 A Simple Java Program 34 1.8 Creating, Compiling, and Executing a Java Program 37 1.9 Programming Style and Documentation 40 1.10 Programming Errors 42 1.11 Developing Java Programs Using NetBeans 45	Lectures	Textbook1 Pages: 23 - 45
2	Elementary Programming 2.1 Introduction 56 2.2 Writing a Simple Program 56 2.3 Reading Input from the Console 59 2.4 Identifiers 62 2.5 Variables 62 2.6 Assignment Statements and Assignment Expressions 64 2.7 Named Constants 65 2.8 Naming Conventions 66 2.9 Numeric Data Types and Operations 67 2.10 Numeric Literals 70 2.11 Evaluating Expressions and Operator Precedence 72 2.12 Case Study: Displaying the Current Time 74 2.13 Augmented Assignment Operators 76	Lectures	Textbook1 Pages: 56 - 79

QFXX/0408-4.0E		Course Plan for Bachelor program - Study Plan Development and Updating Procedures/ Computer Science Department	
	2.14 Increment and Decrement Operators 77		
3	Selections 3.1 Introduction 98 3.2 Boolean Data Type 98 3.3 if Statements 100 3.4 Two-Way if-else Statements 102 3.5 Nested if and Multi-Way if-else Statements 103 3.6 Common Errors and Pitfalls 105 3.7 Generating Random Numbers 109 3.8 Case Study: Computing Body Mass Index 111	Lectures	Textbook1 Pages: 98 – 112 Example from D.S Malik
4	3.9 Case Study: Computing Taxes 112 3.10 Logical Operators 115 3.11 Case Study: Determining Leap Year 119 3.12 Case Study: Lottery 120 3.13 switch Statements 122 3.14 Conditional Operators 125 3.15 Operator Precedence and Associativity 126 3.16 Debugging 128	Lectures	Textbook1 Pages: 112 – 128 Example from D.S Malik
5	Mathematical Functions, Characters, and Strings 4.1 Introduction 142 4.2 Common Mathematical Functions 142 4.3 Character Data Type and Operations 147 4.4 The String Type 152 4.5 Case Studies 161 4.6 Formatting Console Output 167	Lectures	Textbook1 Pages: 142 - 167 Example from D.S Malik
6	Loops 5.1 Introduction 182 5.2 The while Loop 182 5.3 Case Study: Guessing Numbers 185 5.4 Loop Design Strategies 188 5.5 Controlling a Loop with User Confirmation or a Sentinel Value 190 5.6 The do-while Loop 192 5.7 The for Loop 195	Lectures	Textbook1 Pages: 182 - 195 Example from D.S Malik
7	5.8 Which Loop to Use? 198 5.9 Nested Loops 200 5.10 Minimizing Numeric Errors 202 5.11 Case Studies 204 5.12 Keywords <i>break</i> and <i>continue</i> 208 5.13 Case Study: Checking Palindromes 211 5.14 Case Study: Displaying Prime Numbers 213	Lectures	Textbook1 Pages: 198 - 213 Example from D.S Malik
8	Methods 6.1 Introduction 228 6.2 Defining a Method 228 6.3 Calling a Method 230 6.4 void vs. Value-Returning Methods 233 6.5 Passing Parameters by Values 236	Lectures	Textbook1 Pages: 228 – 236 Example from Deitel

QFXX/0408-4.0E	Course Plan for Bachelor program - Study Plan Development and Updating Procedures/ Computer Science Department
----------------	---

	MIDTERM EXAM		
9	6.6 Modularizing Code 239 6.7 Case Study: Converting Hexadecimals to Decimals 241 6.8 Overloading Methods 243 6.9 The Scope of Variables 246 6.10 Case Study: Generating Random Characters 247	Lectures	Textbook1 Pages: 239 - 249 Example from Deitel
10	Revision, Examples and Assignments Homework discussion	Lectures	
11	Single-Dimensional Arrays 7.1 Introduction 270 7.2 Array Basics 270 7.3 Case Study: Analyzing Numbers 277 7.4 Case Study: Deck of Cards 278	Lectures	Textbook1 Pages: 270 – 280 Example from D.S Malik
12	7.5 Copying Arrays 280 7.10 Searching Arrays 289 Examples	Lectures	Textbook1 Pages: 281 - 294
13	Multidimensional Arrays 8.1 Introduction 312 8.2 Two-Dimensional Array Basics 312 8.3 Processing Two-Dimensional Arrays 315	Lectures	Textbook1 Pages: 312 – 315 Example from D.S Malik
14	8.6 Case Study: Finding the Closest Pair 320 8.8 Multidimensional Arrays 325 8.9 Array Revision	Lectures	Textbook1 Pages: 320 - 325
15	<ul style="list-style-type: none"> Programming Examples. Project. Revision. 	Lectures	
16	Final Exam		

* Learning styles: Lecture, flipped learning, learning through projects, learning through problem solving, participatory learning ... etc.

** Reference: Pages in a book, database, recorded lecture, content on the e-learning platform, video, website ... etc.

Schedule of asynchronous interactive activities (in the case of e-learning and blended learning)

Week	Task / activity	Reference	Expected results
1			
2			
3			
4			
5			
6			
7			
8			

QFXX/0408-4.0E	Course Plan for Bachelor program - Study Plan Development and Updating Procedures/ Computer Science Department		
9			
10			
11			
12			
13			
14			
15			
16			