



Course Specifications

Course Title:	Object Oriented Programming
Course Code:	BSCS 2355
Program:	Bachelor of Computer Science
Department:	Computing, and Informatics
College:	Hekma School of Engineering
Institution:	Dar Al-Hekma University

Table of Contents

A. Course Identification.....	3
6. Mode of Instruction (mark all that apply)	3
B. Course Objectives and Learning Outcomes.....	3
1. Course Description	3
2. Course Main Objective.....	3
3. Course Learning Outcomes	4
C. Course Content	4
D. Teaching and Assessment	4
1. Alignment of Course Learning Outcomes with Teaching Strategies and Assessment Methods	4
2. Assessment Tasks for Students	5
E. Student Academic Counseling and Support	6
F. Learning Resources and Facilities.....	6
1. Learning Resources	6
2. Facilities Required.....	6
G. Course Quality Evaluation	7
H. Specification Approval Data	7

A. Course Identification

1. Credit hours:			
2. Course type			
a.	University <input type="checkbox"/>	College <input type="checkbox"/>	Department <input checked="" type="checkbox"/>
b.	Required <input checked="" type="checkbox"/>	Elective <input type="checkbox"/>	Others <input type="checkbox"/>
3. Level/year at which this course is offered: 2 nd year, 1 st semester			
4. Pre-requisites for this course (if any): BSCS1350 Introduction to Programming			
5. Co-requisites for this course (if any):			

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	30
3	Tutorial	
4	Others (specify)	
	Total	60

B. Course Objectives and Learning Outcomes

1. Course Description

This course covers the concepts of object-oriented programming (OOP). It provides a review of the object-oriented programming paradigm with an emphasis on the definition and use of classes and objects, as well as fundamental OOP concepts like encapsulation & information hiding, inheritance, polymorphism & overriding, as well as abstraction & interfaces. The course offers an introduction to Graphical User Interface (GUI), and also covers fundamental programming concepts including files & streams operations and exception handling.

2. Course Main Objective

A major course that fulfils the requirements of the CS POS for graduation.

The main purpose of the course is to introduce the students to the concepts of object-oriented programming. The students will be able to define classes and subclasses and make use of all the features that classes entails. Students will practice encapsulation and information-hiding to protect and conform proper usage of each class. The students will use other object-oriented design features such as inheritance and polymorphism where class members are allowed to be

utilized elsewhere. Finally the students will be exposed to interfaces and overloading, exception handling, generics, and class collections.

3. Course Learning Outcomes

CLOs		Aligned PLOs
1	Knowledge and Understanding	
1.1		
1.2		
1.3		
1...		
2	Skills:	
2.1	Implement Encapsulation and Information Hiding	2
2.2	Use abstract classes and interfaces.	2
2.3	Apply polymorphism and overriding techniques.	2
2.4	Develop advanced OO programs using inheritance.	2
2.5	Use exception-handling techniques to manage exceptions.	2
2.6	Construct programs using I/O stream operations.	2
2.7	Trace different programs to identify errors or produce outputs.	2
2.8	Develop programs with a Graphical User Interface (GUI)	2
3	Values:	
3.1	Implement a fully functioning project in teams	5
3.2		
3.3		
3...		

C. Course Content

No	List of Topics	Contact Hours
1	Classes and Objects	4
2	Encapsulation and Information Hiding	6
3	Abstraction and Interfaces	4
4	Polymorphism and Overloading	8
5	Inheritance and Overriding	10
6	Exception Handling	8
7	Files and Streams Operations	8
8	Graphical User Interface (GUI)	12
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		
2.0	Skills		
2.1	Implement encapsulation and information hiding.	<ul style="list-style-type: none"> • Lectures • Labs 	<ul style="list-style-type: none"> • Quizzes • Assignments

Code	Course Learning Outcomes	Teaching Strategies	Assessment Methods
			<ul style="list-style-type: none"> Exams Project and Presentation
2.2	Use abstract classes and interfaces.	<ul style="list-style-type: none"> Lectures Labs 	<ul style="list-style-type: none"> Quizzes Assignments Exams Project and Presentation
2.3	Apply polymorphism and overriding techniques.	<ul style="list-style-type: none"> Lectures Labs 	<ul style="list-style-type: none"> Quizzes Assignments Exams Project and Presentation
2.4	Develop advanced OO programs using inheritance.	<ul style="list-style-type: none"> Lectures Labs 	<ul style="list-style-type: none"> Quizzes Assignments Exams Project and Presentation
2.5	Use exception-handling techniques to manage exceptions.	<ul style="list-style-type: none"> Lectures Labs 	<ul style="list-style-type: none"> Quizzes Assignments Exams Project and Presentation
2.6	Construct programs using I/O stream operations.	<ul style="list-style-type: none"> Lectures Labs 	<ul style="list-style-type: none"> Quizzes Assignments Exams Project and Presentation
2.7	Trace different programs to identify errors or produce outputs.	<ul style="list-style-type: none"> Lectures Labs 	<ul style="list-style-type: none"> Quizzes Exams
2.8	Develop programs with a Graphical User Interface (GUI)	<ul style="list-style-type: none"> Lectures Labs Project 	<ul style="list-style-type: none"> Quizzes Exams
3.0	Values		
3.1	Implement a fully functioning project in teams.		
3.2			
...			

2. Assessment Tasks for Students

#	Assessment task*	Week Due	Percentage of Total Assessment Score
1	Assignment I	4 th	10
2	Quizzes	5 th	15
4	Midterm	9 th	20
6	Project	15 th	15
7	Final Exam	17 th	30
8	Participation		10

#	Assessment task*	Week Due	Percentage of Total Assessment Score
8			

*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 :

Full time faculty are required to have a minimum of 10 office hours per week on campus.

Usually, the time allotted to student exceeds this amount since faculty are always available to students as required.

Part time faculty are required to have a minimum of one office hour per week on campus for each course. The faculty is also available through email and Blackboard messaging system.

F. Learning Resources and Facilities

1. Learning Resources

Required Textbooks	Phillips, D. (2018). <i>Python 3 Object-Oriented Programming. 3rd Edition</i> . Birmingham, UK: Packt Publishing. ISBN-13: 978-1-849511-26-1
Essential References Materials	
Electronic Materials	Cisco PCAP: Programming Essentials In Python
Other Learning Materials	Gaddis, T. (2017). <i>Starting Out with Python. 4th Edition</i> . Essex, UK: Pearson Education Limited. ISBN-13: 978-0134444321

2. Facilities Required

Item	Resources
Accommodation (Classrooms, laboratories, demonstration rooms/labs, etc.)	Classrooms and Labs
Technology Resources (AV, data show, Smart Board, software, etc.)	A. Python 3 B. A suitable IDE and code editor
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
Course Indirect Assessment (ABET): the instructor collects valuable feedback regarding the course CLOs	Students	Course Survey, indirect
Student course evaluations: The university collects valuable feedback from the student course evaluation which is completed at the end of the semester for each course.	Students	University Survey, indirect
Other surveys: The University gathers several surveys measuring teaching effectiveness; this includes Student Satisfaction Survey and Graduating Senior Survey which are both held every year.	Students	Questionnaire, indirect
Peer & department chair visits and evaluations	Faculty members	Visits & evaluation form, indirect
Performance management KPIs annual assessment	Quality Assurance Office	Forms, direct
Course reports	Faculty Members	Forms, direct
Annual program reports	Program Director	Forms, direct
External evaluation for course reports and files once a year	External examiner	Forms, indirect

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	6
Reference No.	6
Date	20/05/2021