

1.	Course Title	Object Oriented Programming		
2.	Course Code	CCPS1523		
3.	Status	Major		
4.	Credit Hour	3 (2+1) 2 lecture (2 hours lecture x 14 weeks) 1 tutorials (1.5 hours per x 14 weeks) using simulator & emulator supervised by tutor		
5.	Semester/Year	2/1		
6.	Prerequisites	CCPS1513 Computer Programming		
7.	Teaching method:	Distance Learning (Electronic)		
8.	Evaluation	Assessment and Marking Percentage: Participation 5% Quizzes 15% Project 15% Mid Sem Exam 15% Final Examination 50%		
9.	Lecturer			
10.	Objective of the Subject	To provide a good understanding of the key features of object-oriented technology as well as an industry standard methodology (UML) for Object-Oriented Analysis and Design. Students will be expected to analyze and design an Object-Oriented system in UML and implement it using C++		
11.	Learning Outcomes	By the end of the subject, students should be able to: - Understand the key features of object-oriented technology as well as standard methodology (UML) for Object-Oriented Analysis and Design. - Demonstrate the basic notions and techniques for algorithm development - Implement these algorithms using C++ correctly and effectively.		
12.	Synopsis	The major areas of study include: Basic Concepts of Object Oriented Technology, State, Behaviour, and Identity of Objects, Principles in Object-Oriented Programming, Object-Oriented Analysis and Design, Programming in C++ as Object-Oriented Language, and Case Studies		
13.	Topics	Details	Lecture (Hrs)	Tutorial (Hrs)
	Topic 1	Introduction to Object-Oriented Classes and Objects, Abstraction, Encapsulation,	2	1.5
	Topic 2	Inheritance, Polymorphism, Message Passing, OOAD Methodologies, introducing UML	2	3
	Topic 3	Object-Oriented Analysis Syntax and Semantic (with examples) of Use Case Diagrams,	2	1.5
	Topic 4	Package Diagrams, Class Diagrams, Collaboration Diagrams,	2	1.5
	Topic 5	Sequence Diagrams, State Diagrams, Activity Diagrams	2	1.5
	Topic 6	Object-Oriented Design Syntax and Semantic (with examples) of Component Diagrams.	3	1.5
	Topic 7	Object-Oriented Design Syntax and Semantic (with examples) Deployment Diagrams.	3	1.5
	Topic 8	- Implementation in an Object-Oriented Language - Comparing How Object-Oriented Languages (like C++, Java, Eiffel, Smalltalk),	2	2
	Topic 9	Implement Object-Oriented Concepts.	2	2

Bachelor of Information Technology (Hons)

	Topic 10	Case Studies Analyzing, Designing a Business Information System using UML and Implementing it Using a Chosen OOPL (Preferably C++).	4	3
	Topic 11	Analyzing, Designing a Real-Time or Control System using UML and Implementing it Using a Chosen OOPL (Preferably C++).	4	2
		Total contact hours	28	21
		Equivalent lecture hours	28	14
		Total lecture hours	42	
		Credit hours	3	
14.	Main reference: Textbook:	- Grady Booch, James Rumbaugh, and Ivar Jacobson, Unified Modeling Language User Guide, The (2nd Edition) (Addison-Wesley Object Technology Series) (2005) - Joyce Farrell, Object-Oriented Programming Using C++ (2008)		
15.	Additional References:	- John W. Satzinger, Robert B. Jackson, and Stephen D. Burd, Object-Oriented Analysis and Design with the Unified Process (2004) - James Rumbaugh, Ivar Jacobson, Grady Booch:“ Unified Modeling Language Reference Manual,The (2nd Edition) (Addison-Wesley Object Technology Series)”, Addison-Wesley, 2004 - Ivar Jacobson, Grady Booch, James Rumbaugh: “The Unified Software Development Process”, Addison-Wesley,1999 - Martin Fowler, Kendall Scott: “UML Distilled: A Brief Guide to the Standard Object Modeling Language, Third Edition”, Addison-Wesley, 2003 H. Eriksson, M. Penker, “UML Toolkit”, John Wiley & Sons, 1998.		
	Other Materials:	All materials will be available to the students online.		