

1.	Name of Course/Module	Object Oriented Programming	
2.	Course Code	TCP1311	
3.	Status of Subject	Core for B.IT Artificial Intelligence	
4.	MQF Level/Stage	Bachelor Degree – MQF Level 6	
5.	Version (state the date of the last Senate approval)	June 2012	
6.	Requirement for Registration	Computer Programming II (TCP1241)	
7.	Name(s) of academic/teaching staff	Md. Shohel Sayeed Cheah Wooi Ping Chong Lee Ying	
8.	Semester and Year offered	Trimester 1 (Gamma Level)	
9.	Objective of the course/module in the programme :		
	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 Java.		
10.	Learning Outcomes :		
	At the completion of the subject, students should be able to:		
	LO1: Describe the key features of object-oriented technology (Cognitive, Level 1)		
	LO2: Explain different programming structures and executions for both standalone and applet application (Cognitive, Level 3)		
	LO3: Describe the representation of an industry standard methodology (UML) for Object-Oriented Analysis and Design (Cognitive, Level 2)		
	LO4: Demonstrate the basic notions and techniques for algorithm development (Cognitive, Level 3)		
	LO5: Develop both standalone and applet program (Cognitive, Level 5)		
11.	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 Java as Object-Oriented Language, and Case Studies		
	Bidang pengajian meliputi: Konsep Asas Teknologi Berorientasi Objek, Keadaan, Ciri-ciri, dan Identiti Objek, Prinsip-prinsip Pengaturcaraan Berorientasi Objek, Analisa dan Rekabentuk Berorientasi Objek, Pengaturcaraan Java sebagai Bahasa Berorientasi Objek, dan Kajian Kes.		
12.	Mapping of Subject to Programme Outcomes :		
	Programme Outcomes		<b>% of Contribution</b>
	PO1: Apply soft skills in work and career related activities		12.5
	PO2: Demonstrate knowledge and understanding of fundamental concepts, principles and best practices		37.5
	PO3: Analyse the requirements to address problems or opportunities in relevant domains or organisations		25.0
	PO5: Blend innovative mind and entrepreneurial skills		25.0
13.	Assessment Methods and Types :		
	Method and Type	Description/Details	Percentage
	Mid Term Test	Written Exam	20%

	Lab Test	Practical Exam	20%
	Assignment	Report & Presentation	20%
	Final Exam	Written Exam	40%
14.	Details of Subject		
	Topics	Mode of Delivery	
		Lecture	Lab
	<b>1. Introduction to Object-Orientation</b> Classes and Objects, Abstraction, Encapsulation, Inheritance, Polymorphism, Message Passing, OOAD Methodologies, introducing UML	4	4
	<b>2. Object-Oriented Analysis</b> Syntax and Semantic (with examples) of Use Case Diagrams, Package Diagrams, Class Diagrams, Collaboration Diagrams, Sequence Diagrams, State Diagrams, Activity Diagrams.	6	6
	<b>3. Object-Oriented Design</b> Syntax and Semantic (with examples) of Component Diagrams and Deployment Diagrams.	6	6
	<b>4. Implementation in an Object-Oriented Language</b> Comparing How Object-Oriented Languages (like C++, Java, Eiffel, Smalltalk), implement Object-Oriented Concepts.	4	4
	<b>5. Case Studies</b> Analyzing, Designing two Systems (A Business Information System and a Real-Time or Control System) using UML and Implementing Them Using a Chosen OOPL (Preferably Java).	8	8
	<b>Total</b>	<b>28</b>	<b>28</b>
15.	Laboratory/Practical		
	Students will be working programming exercises based on topics covered in the week.		
16.	Total Student Learning Time (SLT)	Face to Face (Hour)	Total Guided and Independent Learning
	Lecture	28	28
	Tutorials		
	Laboratory/Practical	28	14
	Presentation		
	Assignment	-	10
	Mid Term Test	1	5
	Lab Test	2	4
	Final Exam	2	15
	Sub Total	61	76
	Total SLT	137	
17.	Credit Value	$137/40 = 3.425 \Rightarrow 3$	
18.	Reading Materials :		
	Textbook	Reference Materials	

	<ol style="list-style-type: none"> <li>1. Java Software Solutions, 6th Edition, John Lewis and William Loftus, 2009, Pearson International Edition.</li> </ol>	<ol style="list-style-type: none"> <li>2. James Rumbaugh, Ivar Jacobson, Grady Booch: "Unified Modeling Language Reference Manual, The (2nd Edition) (Addison-Wesley Object Technology Series)", Addison-Wesley, 2004</li> <li>3. Ivar Jacobson, Grady Booch, James Rumbaugh: "The Unified Software Development Process", Addison-Wesley, 1999</li> <li>4. Martin Fowler, Kendall Scott: "UML Distilled: A Brief Guide to the Standard Object Modeling Language, Third Edition", Addison-Wesley, 2003</li> <li>5. H. Eriksson, M. Penker, "UML Toolkit", John Wiley &amp; Sons, 1998.</li> <li>6. H. Eriksson, M. Penker, Brian Lyons, David Fado "UML 2 Toolkit", John Wiley &amp; Sons, 2003.</li> <li>7. J. Rumbaugh, M. Blaha, W. Premerlani, F. Eddy, W. Lorensen, "Object-Oriented Modelling and Design", Prentice Hall, 1991.</li> <li>8. J. Rumbaugh, M. Blaha, "Object-Oriented Modeling and Design with UML (2nd Edition)", Prentice Hall, 2004.</li> <li>9. V. Jacobson, "Object-Oriented Software Engineering", Addison Wesley, 1992.</li> <li>10. G. Booch, "Object-Oriented Analysis and Design with Applications", Benjamin Cummings, 3/E 2004</li> <li>11. Grady Booch, James Rumbaugh, Ivar Jacobson: "The Unified Modeling Language User Guide", Addison-Wesley, 1998.</li> </ol>
19.	<p>Appendix (to be compiled when submitting the complete syllabus for the programme) :</p> <ol style="list-style-type: none"> <li>1. Mission and Vision of the University and Faculty</li> <li>2. Mapping of Programme Objectives to Vision and Mission of Faculty and University</li> <li>3. Mapping of Programme Outcome to Programme Objectives</li> <li>4. Programme Objective and Outcomes (Measurement and Descriptions)</li> </ol>	