MULTIMEDIA UNIVERSITY®

COURSE INFORMATION

1 .	**Name of Course**	Object Oriented Programming
2 .	**Course Code**	DCS5088
3 .	**Type of Course** **(e.g. : Core, major, elective etc.)**	Core
4 .	**Synopsis**	This subject covers the basic and advanced object oriented programming concepts in order to solve simple to moderate problems. The course covers topics such as C++ fundamentals, classes and objects, constructors, destructors and friends, dynamic memory allocation, inheritance, function and operator overloading and lastly virtual functions.
5 .	**Version** **(State the date of theSenate's approval - previous and the current approval date)**	Current: October 2017 Previous: June 2017
6 .	**Name(s) of Academic Staff**	Julie Yew, Nurul Aini Nordan, Usha Vellappan, Nun Shwu Huey
7 .	**Semester and Year Offered**	Year 1 Trimester 3
8 .	**Credit Value**	4
9 .	**Pre-Requisite**	DCS5038 Program Design

10 . **Objective of the course in the programme:**

To initiate skills to write algorithms and programs. Students will be introduced to current programming paradigms, programming methodology and object – oriented programming.

11 . **Justification for including the course in the programme:**

The subject is offered to expose students to object oriented programming methodology as virtually all programming languages, scripting languages and application designs are object-oriented or object-based. Therefore, it is crucial that students be familiar with object oriented programming methodology.

14 . **Transferable Skills:**

Practical skills and problem solving skills. Teamwork, communication and leadership skills.

15 . **Distribution of Student Learning Time (SLT)**

	Course Content Outline	**CLO	Teaching and Learning Activities – Guided Learning (F2F)* *L	*T	*P	*O	Guided Learning (NF2F)*	Independent Learning (NF2F)*	Total SLT
1	**Introduction** Comparison of procedure oriented, structure oriented and object-oriented programming paradigms, top-down design, algorithm development, refining algorithms, fundamental of object oriented design, identifying, attributes and behaviour, features of object oriented programming, object models.	**1,2**	2					2	4
2	**C++ Fundamentals** Equations, Standard I/O streams, function prototypes, C++ enhancements to C, default function parameters, inline functions, reference variables, comparison between pointers and references.	**1,2**	7		4		3	6	20
3	**Classes and Objects** Creating new data type in C++, class declaration, members, access functions, constant objects, member objects, static members, arrays of class objects	**1,2,3**	8		6		2	7.5	23.5
4	**Constructors, Destructors and Friends** Constructors and destructors, copy constructor, friend functions, classes as friends, object composition	**1,2,3**	6		6		2	5.5	19.5
5	**Dynamic Memory Allocation** Free store, new and delete operators, class with pointer members, this pointer and constant member functions, assignment vs initialization, passing and returning objects	**1,2,3**	5		4		1	5	15
6	**Inheritance** Protected members, handling related types in C++, derived classes, conversion between base and derived classes, single inheritance, multiple inheritance, passing arguments to base class constructor	**1,2,3**	6		4		2	5	17
7	**Function and Operator Overloading** Function overloading, operator overloading	**1,2,3**	3		2		2	2	9

	8	**Virtual Functions** Virtual functions, dynamic binding, pure virtual function, abstract base classes	**1,2,3**	3					3	6
								Total SLT	**114**

SUMMATIVE ASSESSMENT

1. Continuous Assessment	**Percentage %**	**Total SLT**
Lab	10%	5
Quiz	10%	3
Assignment	5%	4
Project	10%	6
Midterm	15%	6
Total SLT for Continuous Assessment		**24**

2. Final Assessment	**Percentage %**	**Total SLT**	
		F2F	**ILT**
Final Exam		2	20
Total SLT for Final Assessment (F2F + NF2F)		**22**	
Grand Total	**50%**	**160**	

****Indicate the CLO based on the CLO's numbering in Item 12.**
***L= Lecture, *T= Tutorial, *P= Practical, *O= Others, F2F*= Face to Face, NF2F*= Non Face to Face**

16 . **Identify Special Requirement to Deliver the Course (e.g., software, nursery, computer lab, simulation room):**
Software Dev C++, Online compiler, CodeBlocks

17 . **Main References:**
Gaddis, T., Walters, J., & Muganda, G. (2016). *Starting out with C++: early objects (9th ed.)* . Addison Wesley.

18 . **Additional References:**
1. Deitel, P., & Deitel, H. (2017). *C++ how to program: early objects version (10th ed.)* . Prentice Hall.
2. Dale, N. (2013). *Programming and Problem Solving with C++: Comprehensive Edition, (6th ed.)* . Jones & Bartlett Publishers.
3. Malik, D. S. (2017). *C++ Programming: from problem analysis to program design (7th ed.).* Cengage Learning.