

|     |   |   |
|-----|---|---|
| 1.  | Name of Course/Module   | Computer Programming I                                      |
| 2.  | Course Code   | TCP1231   |
| 3.  | Status of Subject   | Core for B.IT Artificial Intelligence                       |
| 4.  | MQF Level/Stage   | Bachelor Degree – MQF Level 6                               |
| 5.  | Version<br>(state the date of the last Senate approval)   | June 2012   |
| 6.  | Requirement for Registration  | None  |
| 7.  | Name(s) of academic/teaching staff  | Cheah Wooi Ping<br>Md. Shohe! Sayeed<br>Goh Kah Ong Michael |
| 8.  | Semester and Year offered   | Trimester 1 (Beta Level)                                    |
| 9.  | Objective of the course/module in the programme :   |   |
|     | To give an introduction to basic programming concepts through the use of a high-level programming language such as C/C++. It covers the basic notions and techniques for algorithm development and the implementation of algorithms in a high-level programming language.   |   |
| 10. | Learning Outcomes :   |   |
|     | At the completion of the subject, students should be able to:   |   |
|     | LO1: Identify basic structures of a high level programming language correctly (Cognitive, Level 1)<br>LO2: Demonstrate the basic notions and techniques for algorithm development (Cognitive, Level 3)<br>LO3: Apply basic concepts of a high level programming language correctly (Cognitive, Level 3)<br>LO4: Develop program in a high-level programming language correctly and effectively (Cognitive, Level 5)   |   |
| 11. | Synopsis:   |   |
|     | The major areas of study include: Software Development Life Cycle, Top-down Design, Program Design Steps and Programming Methodology, Structure Chart, Flowchart, Pseudo Code, Debugging and Documentation Techniques, Identifiers, Data Types, Operators, Various Statements, Type Conversion, Conditional and Control Structures, Functions, Arrays and Pointers, Strings, Structures and Unions, File Handling, Command Line Parameters, Pointers to Functions, Header Files, Stacks, Linked Lists, Bitwise Manipulation, and Programming laboratory exercises.                |   |
|     | Bidang pengajian meliputi: Kitaran hayat pembangunan perisian, Rekabentuk atas-bawah, Kaedah rekabentuk aturcara dan pengaturcaraan, Carta struktur, Carta alir, Kod pembayang, Teknik penyahpejatan dan dokumentasi, Pengenalpasti, Jenis-jenis data, Pengendali, Pelbagai kenyataan, Penukaran jenis, Struktur bersyarat dan struktur kawalan, Fungsi, Tatasusunan dan penunjuk, Rangkaian, Struktur dan gabungan, Penggunaan fail, Parameter baris perintah, Penunjuk ke fungsi, Fail pengepala, Timbunan, Senarai berpaut, Manipulasi bit, dan Latihan pengaturcaraan makmal. |   |

|     |   |                     |                          |
|-----|---|---------------------|--------------------------|
| 12. | Mapping of Subject to Programme Outcomes :  |                     |                          |
|     | Programme Outcomes  |                     | <b>% of Contribution</b> |
|     | PO1: Apply soft skills in work and career related activities  |                     | 50                       |
|     | PO2: Demonstrate knowledge and understanding of fundamental concepts, principles and best practices   |                     | 50                       |
| 13. | Assessment Methods and Types :  |                     |                          |
|     | Method and Type   | Description/Details | Percentage               |
|     | Mid Term Test   |                     | 20%                      |
|     | Assignments   |                     | 20%                      |
|     | Laboratory / Practical  |                     | 20%                      |
|     | Final Exam  |                     | 40%                      |
| 14. | Details of Subject  |                     |                          |
|     | Topics  | Mode of Delivery    |                          |
|     |   | Lecture             | Lab                      |
|     | <b>1. Software Development and Programming Environment</b><br>Software development life cycle - Top-down design with function system structure - program design steps and programming methodology - structure chart - flowchart - pseudo code – Debugging and documentation techniques. | 3                   | 3                        |
|     | <b>2. Syntactic Structure of a Program</b><br>Identifiers - data types – operators - various statements - operator precedence - type conversion – conditional and control structures - function - recursive functions.  | 4                   | 4                        |
|     | <b>3. Arrays</b><br>One- and two-dimensional arrays - passing individual elements or whole array to a function - Simple sorting and searching on arrays - pointers - strings - dynamic memory allocation.   | 6                   | 6                        |
|     | <b>4. Structures and Unions</b><br>Structure declaration and definition – accessing structures - array of structures - pointers and structures – Union declaration – enumerated variables.  | 5                   | 5                        |
|     | <b>5. File Handling</b><br>Concept of a file - files and streams – standard file handling functions - binary files - random access files.   | 5                   | 5                        |
|     | <b>6. Advanced Topics</b><br>Command line parameters – pointers to functions - creation of header files - stacks - linked lists – bitwise manipulation  | 5                   | 5                        |
|     | <b>Total</b>  | <b>28</b>           | <b>28</b>                |
| 15. | Tutorials   |                     |                          |
|     | <ul style="list-style-type: none"> <li>Students will be working in programming exercises based on relevant topics covered in the corresponding week.</li> </ul>   |                     |                          |

|     |  |  |                                       |
|-----|--|--|---------------------------------------|
| 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  | 4                                     |
|     | Final Exam   | 2  | 15                                    |
|     | Lab Test   | 1  | 2                                     |
|     | Sub Total  | 60   | 73                                    |
|     | Total SLT  | 133/40 = 3.3 => 3  |                                       |
| 17. | Credit Value   | 3  |                                       |
| 18. | Reading Materials :  |  |                                       |
|     | Textbook   | Reference Materials  |                                       |
|     | 1. Walter Savitch, <b>Problem Solving with C++</b> , Fourth Edition, Addison Wesley, 2004.   | 1. Goran Svenk, <b>Object Oriented Programming using C++ for Engineering and Technology</b> , Thomson publishing, 2003.<br>2. Walter Savitch, <b>Problem Solving: The object of programming</b> , Fourth Edition, Addison Wesley, 2004.<br>3. D. S. Malik, <b>C++ programming</b> , Second Edition, Thomson Publishing, 2004.<br>4. Ira Pohl, <b>C++ by Dissection</b> , Addison Wesley, 2003. |                                       |
| 19. | Appendix (to be compiled when submitting the complete syllabus for the programme) :  |  |                                       |
|     | 1. Mission and Vision of the University and Faculty<br>2. Mapping of Programme Objectives to Vision and Mission of Faculty and University<br>3. Mapping of Programme Outcome to Programme Objectives<br>4. Programme Objective and Outcomes (Measurement and Descriptions) |  |                                       |