**SOFTWARE DEVELOPMENT: Outcome Key Knowledge Textbook Pages**

**Unit 3: Outcome 1 – Programming Practice**

On completion of this unit the student should be able to interpret designs and apply a range of functions and techniques using a programming language to develop working modules.

|  |  |
| --- | --- |
| **Data and information** |  |
| **1.1 Problem solving methodology** | **Ch1:5-11** |
| 1.2 characteristics of data types | Ch7:79-80 |
| 1.3 types of data structures, including one-dimensional arrays (single data type, integer index) and records (varying data types, field index) | Ch7:80-82 |
| **Approaches to problem solving** |  |
| 1.4 methods of representing designs, including data dictionaries, object descriptions, mock-ups and pseudocode | Ch5:57-60 |
| formatting and structural characteristics of input and output, including XML file formats and CSV (text file) | Ch8:100-101 |
| a programming language as a method for developing working modules that meet specific needs | Ch6:71-73 |
| processing features of a programming language, including instructions, procedures, methods, functions and control structures | Ch6:71-73 |
| techniques for linear and binary searching | Ch7:84-87 |
| techniques for checking that modules meet design specifications, including trace tables and test data | Ch9:111-113 |
| purposes and characteristics of internal documentation, including comments and meaningful names. | Ch6:73-74 |

**Unit 3: Outcome 2 – Analysis and Design**

On completion of this unit the student should be able to analyse and document a need or opportunity, generate alternative design ideas, represent the preferred solution design and formulate a project plan for creating the solution.

|  |  |
| --- | --- |
| **Data and information** |  |
| techniques for collecting data to determine needs and requirements, including interviews, surveys and observation | Ch3:25-27 |
| **Approaches to problem solving** |  |
| features of functional and non-functional requirements | Ch3:27-28 |
| constraints that influence solutions, including economic, legal, social, technical and useability factors | Ch3:28-30 |
| factors that determine the scope of solutions | Ch3:30-31 |
| features and purposes of software requirements specifications | Ch3:31-32 |
| techniques for generating design ideas | Ch5:53-56 |
| criteria for evaluating alternative design ideas and the efficiency and effectiveness of solutions | Ch5:61-64 |
| tools and techniques for depicting the interfaces between solutions, users and networks, including use case diagrams created using Unified Modelling Language | Ch4:37-40 |
| features of context diagrams and data flow diagrams | Ch4:41-50 |
| methods of expressing software designs using data dictionaries, object descriptions, mock-ups and pseudocode | Ch5:56-61 |
| factors influencing the design of solutions, including useability, affordability, security, interoperability and marketability | Ch5:64-65 |
| characteristics of user experiences, including efficient and effective user interfaces | Ch5:66-68 |
| naming conventions for solution elements | Ch6:74 |
| project management concepts and processes, including milestones and dependencies (concepts), and task identification, sequencing, time allocation, resources and documentation using Gantt charts (processes) | Ch3:32-34 |
| **Digital systems** |  |
| security considerations influencing the design of solutions, including data protection and authentication | Ch5:64 |
| styles of modern application architecture, including mobile, rich client, peer-to-peer and internet applications | Ch6:75-76 |
| **Interactions and impact** |  |
| types of goals and objectives of organisations and information systems | Ch1:2-5 |
| key legal requirements relating to the ownership and privacy of data and information. | Ch10:122-130 |

.

**Unit 4: Outcome 1 – Software Solutions**

On completion of this unit the student should be able to apply stages of the problem-solving methodology to create a solution using a programming language that fulfils identified requirements and assess the effectiveness of the project plan in monitoring progress.

|  |  |
| --- | --- |
| **Data and information** |  |
| ways in which file size, storage medium and organisation of files affect access of data | Ch8:106 |
| uses of data structures to organise and manipulate data, including associative arrays (or dictionaries or hash tables) | Ch7:83-84 |
| **Digital systems** |  |
| procedures and techniques for handling and managing files, including security, archiving, backing up and disposing of files | Ch8:100-105 |
| **Approaches to problem solving** |  |
| processing features of a programming language, including instructions, procedures, methods, functions and control structures | Ch6:71-73 |
| algorithms for sorting, including selection sort and quick sort and their suitability for a given purpose, measured in terms of algorithm complexity and sort time | Ch7:88-94 |
| characteristics of efficient and effective solutions | Ch5:61-64 |
| techniques for checking that coded solutions meet design specifications, including construction of test data | Ch9:109-113 |
| validation techniques, including existence checking, range checking and type checking | Ch8:98-99 |
| techniques for testing the useability of solutions and forms of documenting test results | Ch9:113-114 |
| techniques for recording the progress of projects, including annotations, adjustments to tasks and timeframes, and logs | Ch3:32-34 |
| factors that influence the effectiveness of project plans | Ch9:117-118 |
| strategies for evaluating the efficiency and effectiveness of solutions and project plans. | Ch9:115-118 |
|  |  |
|  |  |

**Unit 4: Outcome 2 – Interactions and Impacts**

On completion of this unit the student should be able to analyse and explain the dependencies between two information systems and evaluate the controls in place in one information system to protect the integrity of its source data.

|  |  |
| --- | --- |
| **Interactions and impact** |  |
| reasons why individuals and organisations use information systems | Ch1:2-5 |
| goals and objectives of information systems | Ch1:2-5 |
| types of interactions (inputs and outputs) generated by information systems | Ch1:2-5 |
| characteristics of data that has integrity, including accuracy, timeliness, reasonableness, authenticity, correctness | Ch8:99-100 |
| key legislation that affects how organisations control the storage, communication and disposal of their data and information: the *Privacy Act 1988*, the *Privacy and Data Protection Act 2014*, the *Copyright Act 1968*, the *Spam Act 2003* and the *Charter of Human Rights and Responsibilities Act 2006* | Chp10:121-130 |
| data management practices that cause conflict between information systems, including data mining | Ch11:146-147 |
| advantages and disadvantages for stakeholders affected by the operation of information systems | Ch11:148 |
| the impact of diminished data integrity on dependent systems | Ch11:148 |
| **Digital systems** |  |
| the technical underpinnings of intranets, the internet and virtual private networks | Ch2:14-22 |
| characteristics of wired and wireless networks | Ch2:14-22 |
| types and causes of accidental, deliberate and events-based threats to the integrity and security of data and information shared between information systems | Ch11:133-139 |
| the physical and software controls used by organisations to secure the storage and communication of data in a networked environment | Ch11:142-145 |
| the role of hardware, software and technical protocols in managing, controlling and securing data shared between information systems | Ch11:139-142 |
| tools and techniques for tracing transactions between users of information systems. | Ch11:145 |