| | |
|---|---|
| Module Title | **Developing Applications** |
| Level | 5 |
| Reference No. | CSI_5_APD |
| Credit Value | 20 |
| Student Study Hours | Total: 200<br>Contact hours: 52<br>Student managed learning hours: 148 |
| Pre-requisites | Software Development |
| Co-requisites | None |
| Excluded combinations | None |
| Module co-ordinator | TBC |
| Division | Division of Computer Science and Informatics |
| Short Description | This module aims to provide understanding of programming languages and their uses in different application areas. It is using the programming skills/knowledge gained during the first year and explores what kind of applications can be built and for what purpose. |
| Aims | Building on the programming from the first year, the aim of this module is to extend the programming skills through practical work. The module also aims to raise the awareness of different application areas and to give practical experience of developing different applications and providing solutions to different problems. |
| Learning Outcomes | **LO1: Knowledge and Understanding**<br>● Evaluate suitability of programming solutions for different areas of application (Maps to: BCS 2.2.1 a1-a4)<br>**LO2: Intellectual Skills**<br>● Interpret and analyse requirements (Maps to: BCS 2.2.1 a5, a7-a9)<br>**LO3: Practical Skills**<br>● Specify, design, develop, test, correct and document software to implement given requirements (Maps to: BCS 2.2.1 b1-b4)<br>**LO4: Transferable Skills**<br>● Develop problem solving skills (Maps to: BCS 2.2.1 c1, c2) |
| Employability | Programming and Software Development is beginning to be taught more widely. The module is raising awareness of different areas of application such as business, science, art, media, networking, programming for web etc. and gives the experience of developing software in some of these areas. Therefore, it widens the job opportunities for the students. |
| Teaching and learning pattern | The lectures deliver the underpinning knowledge, and the tutorials cover practical work. The learning exercises are there to help you learn new concepts that you need in order to build applications/solutions.<br>The lab sessions allow the tutor to support the students and provide feedback on the provided exercises. |
| Indicative content | ● build on the concepts covered at level 4 and take it to higher level of complexity<br>● new concepts covered depending on the chosen areas of the application, e.g.:<br>● cloud computing<br>● writing APIs<br>● interacting with the database |

| | |
|---|---|
| | ● data analysis<br>● writing programs for Raspberry PI<br>● Web frameworks<br>● Multimedia computation |
| Assessment<br>*Elements & weightings* | **COURSEWORK 100%**<br>**Summative Assessment**<br>Coursework: Expected to consist of individual practical assessments linked to the development of software and an assessment to check the underpinning knowledge and understanding of the development process and the quality of the chosen solution.<br>(LO1- LO4)<br>(Maps to: BCS 2.2.1 a1-a5, a7-a9; b1-b4; c1-c2;)<br>**Formative Assessment**<br>Skills for the summative assessment will be embedded throughout formative opportunities in Lectures and Workshops. Formative assessment will take different forms, such as:<br>● interactive revision quizzes<br>● verbal feedback on tutorial activities<br>● observation and questioning to provide instant feedback as the student takes part in learning activities |
| Indicative Sources<br>*(Reading lists)* | **Core**:<br>● Steven F. Lott, Functional Python programming, Packt Publishing, 2018<br>● Chun, Wesley J., *Core Python Applications programming*, 3rd Edition, Pearson Education, United States, 2012.<br>● http://sthurlow.com/python/<br>● https://en.wikibooks.org/wiki/Non-Programmer%27s_Tutorial_for_Python_3<br>● https://docs.python.org/3/tutorial/index.html - The official python site<br>● There are many other internet resources, and these will be documented on the VLE<br>**Optional:**<br>● “Python Programming for the Absolute Beginner”, Michael Dawson, 3rd Ed, Cengage Learning, 2010.<br>● “Learning Python” Mark Lutz 4th edition O'Reilly.<br>● “Core Python programming”, Wesley. J. Chun 2nd Ed, Prentice Hall, 2007<br>● “Head First Python” Paul Barry<br>● “Programming Python, Second Edition By Mark Lutz<br>● Bell Douglas, Software Engineering for Students: A Programming Approach Addison-Wesley, 4th<br>● Edition, 2005, ISBN 0321261275 |