Module Guide

Fundamentals of Software Development

CSI_4_FSD

School of Engineering

Level 4

Table of Contents

1. MODULE DETAILS

Module Title:	Fundamentals of Software Development
Module Level:	4
Module Reference Number:	CSI_4_FSD
Credit Value:	20
Student Study Hours:	200
Contact Hours:	65
Private Study Hours:	135
Pre-requisite Learning (If applicable):	None
Co-requisite Modules (If applicable):	None
Course(s):	B.Sc. Computer Science
Year and Semester	Year 1 Semester 1
Module Coordinator:	Brahim El Boudani
MC Contact Details (Tel, Email, Room)	elboudab@lsbu.ac.uk
Teaching Team & Contact Details (If applicable):	TBA
Subject Area:	Computer Science and Informatics
Summary of Assessment Method:	100% Coursework
External Examiner appointed for module:	TBC

2. SHORT DESCRIPTION

Software Development skills are widely valued and the skills needed to develop systems requires not only that you understand the problem fully, but that you can work with different technologies, selecting the most appropriate for the task in hand.

Many of the skills acquired in the module are needed later in your course. You will learn different development tools and approaches to solving problems. The more you can take with you from this module into the future, the more you will enrich your later learning. This module aims to begin laying the ground-work for your future success in the field of computer science.

Let's be clear here, just remembering what you are told is not enough. You have to learn to apply what you have been told to new and possibly unfamiliar situations. Finding the answer to the set challenges can be very rewarding.
This module teaches the fundamentals of computer programming covering algorithms, variables, datatypes, arrays, conditional and iterative code and the use of arrays and functions. Students will learn to write simple programs making use of a C programming language and development environment.

3. AIMS OF THE MODULE

1. To provide students with fundamental programming skills.
2. To give students understanding of the fundamental concepts of software development such as source code and the compilation and execution of programs.
3. To enable students to become familiar with the development tools and environments.

4. LEARNING OUTCOMES

4.1 Knowledge and Understanding

You will be able to:

- Describe design notations, software development environments and programming languages and their purpose and interaction.

4.2 Intellectual Skills

You will be able to:

- Interpret and analyze specifications.
- Make effective use of technical reference materials.

4.3 Practical Skills

You will be able to:

- Design, write, test, correct and document simple software to implement given specifications

4.4 Transferable Skills

You will be able to:

- Keep a coherent, evaluative and reflective log of work produced.

5. ASSESSMENT OF THE MODULE

There will be multiple assessment methods used during the duration of the course:

1. During the weekly lab sessions, few programming exercises will be issued leading to formative assessment. Please make sure you attend all the lab sessions to avoid missing the programming examples taught during the session.
2. **2 in-class tests** (30% each):
   - Students who are **late by more than 15 min or absent** will be given 0 marks unless you have extenuating circumstances.
3. **1 lab based coding assignment** (40% total)
   - Students who are **late by more than 15 min or absent** will be given 0 marks unless you have extenuating circumstances.

Coursework	Additional Information	Issued	Due Date
Test 1 (30%)	The test will be based on lecture/lab contents from Week 2 to Week 6	Week 7 (4.11.19)	
Test 2 (30%)	The test will be based on lecture/lab contents from Week 8 to Week 11	Week 12 (11.12.19	
Lab based coding assignment (40%)	The lab based coding assignment will be a C programming test and each student will need to work out few C programs.	Week 14 (13.01.20)	

Class tests occur in week 7 and again in week 12. These assessments will be open-book. You will be allowed to bring along a reflective log-book with notes and reflections made from all the lectures, tutorials and lab sessions. You will not be allowed to use any other materials and you will not be allowed to use electronic versions without special permission. Tutorial sheets with examples will be handed out in Tuesday lab and exercises will be marked and discusses in Tuesday lab.

6. FEEDBACK

Feedback will normally be given to students 15 working days after the final submission of an assignment or as advised by their module leader.

General feedback, applying to all students, will also be placed on the module VLE site within 15 working days.

7. INTRODUCTION TO STUDYING THE MODULE

7.1 Overview of the Main Content

This module emphasises learning through practical exercises and the development of actual software artefacts. Short lectures will be used to inform the laboratory activities and provide a forum for discussion of issues students have encountered in the in the practical work. The lab sessions will occupy the majority of the contact time and will involve much independent working. Students are required to keep a clear record of the work they have done and are encouraged to experiment and investigate beyond the basic material being taught.

7.2 Overview of Types of Classes

There will be a combination of 1 hour lecture sessions delivering technical, theoretical and contextual information, and 4 hour lab sessions in which students will work through practical exercises. These are guided sessions where the aim is for you tolearn practical skills and to make notes in your log-book. These are sessions where you interact with your tutor, It is also an opportunity for you to talk through the work that you did between the sessions.

7.3 Importance of Student Self-Managed Learning Time

Although much laboratory time will be dedicated to practical exercises, students will need to spend more time in independent study during this course. In addition, the theoretical material delivered in lectures will require independent reading and study in order to grasp it fully and prepare for the examination.
In this module you will acquire valuable skills but they might not come easily. Like someone using a gym, you should not be surprised if you do not develop if you have not done the exercises and put in the time. Additional studying is an essential part of this module

7.4 Employability

This module will give you the necessary programming and problem solving skills to secure software development jobs and required vocabulary to discuss software development with colleagues. All computer science professionals need to understand the process of software development even if they do not directly develop software.

8. THE PROGRAMME OF TEACHING, LEARNING AND ASSESSMENT

SEMESTER 1			
WEEK	**Date**	**TOPIC**	**Coursework**
1	23.09.2019	Induction	
2	30.09.2019	Introduction	
3	07.10.2019	C Basics	
4	14.10.2019	C Basics	
5	21.10.2019	Decision Making	
6	28.10.2019	Loops	
7	04.11.2019	Revision	Test 1
8	11.11.2019	Arrays	
9	18.11.2019	Strings	
10	25.11.2019	Functions	
11	02.12.2019	File I/O	
12	09.12.2019	Revision	Test 2
Winter Break from 16th December to 6th January			
13	06.01.2020	Revision	
14	14.01.2020	Lab based Coding Assignment	

9. STUDENT EVALUATION

10. LEARNING RESOURCES

Reading List:

Core Materials:

1. Kernighan, B. W. and Ritchie, D. M. (1988) *The C programming language*. 2nd ed. Englewood Cliffs, N.J.: Prentice Hall. Available from: http://proquestcombo.safaribooksonline.com/9780133086249
2. Vine, M. and Vine, M. (2015) *C programming for the absolute beginner, third edition*. Third edition. Boston, MA: Cengage Learning. Available from: http://0-proquestcombo.safaribooksonline.com.lispac.lsbu.ac.uk/?uiCode=lsbu&xmlId=97813 05273764

Optional Materials:

1. Griffiths, D. and Griffiths, D. (2012) *Head first C*. 1st ed., Vol. Head first series. Beijing: O'Reilly. Available from: http://0-proquestcombo.safaribooksonline.com.lispac.lsbu.ac.uk/?uiCode=lsbu&xmlId=97814 49335649
2. Gookin, D. (2013) *Beginning Programming with C For Dummies*. For Dummies