

School of Computer Science – Coursework 1 Issue Sheet

Session	Semester	Code
18-19	Autumn	COMP2050
Module Convenor	Nasser Mustafa	

Coursework Name	Understanding and improving code
Weight	25 %
Deliverable (a brief description of what is to be handed-in; e.g. 'software', 'report', 'presentation', etc.)	1000 word report reflecting on the process of understanding third party software including relevant examples, diagrams and explanations.
Format (summary of the technical format of deliverable, e.g. "C source code as zip file", "pdf file, 2000 word max", "ppt file, 10 slides max", etc.)	PDF file, 1000 word max. Word count on front page. This should be in the form of a number of screenshots of problem code along with a brief description of any coding problem, and how you would change it (5 main examples). There should also be a class diagram and explanation (see below).

Issue Date	Issued w/b 22 nd October
Submission Date	Monday 12 th November at 4pm. Standard penalty applies for late submissions.
Submission Mechanism	Electronic submission via Moodle. Late Policy (University of Nottingham default will apply, if blank). Standard policy
Feedback Date	w/b 23 rd November
Feedback Mechanism	Individual comments via grading system on Moodle.

Instructions	You will be given the source code of an existing project - a device manager server for connected cars. This code has been modified to provide lots of opportunities for maintenance and refactoring. Your task is to find and read some and then apply what you learned from your
--------------	---

	<p>reading and the slides or other materials covered during the lectures to portions of the code provided. The expected output of the coursework is a report (1000 words max.) where you describe what you have learnt about good programming practices and object oriented software development to date along with a brief description of on the concepts that you have been asked about.</p> <p>Some tips:</p> <ul style="list-style-type: none"> • It might be a good idea to start with reverse engineering a high level class diagram from the source code provided. Doing this and providing an explanation of how you derived it from the source code (e.g. how you identified the type of relationship between classes) will have a positive impact on your mark. Only providing a class diagram without explanations of how you derived it is not sufficient. • When you pick examples you always need to provide an explanation of why your alteration is an improvement to the project. Just providing examples is not enough. Picking the same kind of examples will not bring you any benefits. You need to look out for different kinds of examples (tackling different types of problems with the source code) <p>IMPORTANT: Make sure you understand what you are writing in your report, do not simply copy/paste large portions of text from existing articles or other resources – as this does not demonstrate your understanding of the topic. You might also run into issues with plagiarism.</p> <p>If you use information from other resources please remember to provide a reference to the website / book you found the information in. A quick guide for correct referencing can be found here: https://www.uhi.ac.uk/en/libraries/how-to/UHI-mini-Studentreferencing-guide-en.pdf</p>
--	---