SET11115/SET11515 Formal Approaches to Software Engineering

Coursework 2015/16

# Car Control System

In this coursework you will be expected to specify, design, implement and test an application that is designed to control a car. **The point of the coursework is to create an abstract model of the behaviour of the car. You do not have to implement how the car actually works.**

A controller for a car is required which will ensure safe operation of the car under certain conditions. Importantly

* If the engine fails, the brakes must be applied
* If the brakes fail, the engine must be stopped
* The engine cannot be started without the key being in the ignition
* The car cannot increase its speed above a maximum speed limit
* If petrol is low then a warning light must be shown
* If there is a problem with the oil (temperature / pressure) then a warning light must be shown

Below is a high level diagram showing the components of the system and how they communicate with each other.



# Deliverables

The final deliverable for your coursework project should consist of the following four parts:

## UML Model

Your submission should consist of a UML model showing the transition from initial requirements analysis using Use Case and Activity Diagrams, to a final design consisting of Class Diagrams and Sequence Diagrams. The UML model should be printed as part of your final submission, and also be incorporated as a modelling project in your final Visual Studio 2012 solution.

Your UML model should consist of the following:

* Use Case Diagrams providing an initial analysis of the requirements, actors, subsystems, and functions of the system to be implemented.
* Activity Diagrams analysing the high level behaviour of the system.
* Class Diagrams illustrating the static structure of your system design.
* Sequence Diagrams analysing the behaviour of interactions between the various components in your system design.

**Please note: simply reverse engineering your final implementation into a class diagram is not considered a valid submission for this element of the coursework submission, and will result in a mark of 0.**

## B-Method Specification

Your submission should consist of a B-Method specification of the system design that you developed using UML. In particular, you should show the how the class diagrams and sequence diagrams have led to the system specification. Any required changes at this stage of development should be clearly illustrated in your final report.

This element of the submission should consist of your Atelier B project in electronic format (e.g. on a CD or memory stick) and the Abstract Machine Notation specifications printed as part of the final report submission.

Your B-Method specification should consist of the following:

* Initial MACHINE definitions for the components identified in your static structure design, and their behaviours defined in your sequence diagrams and activity diagrams.
* Any REFINES definitions that you may have produced during the development process.
* The proof obligations for all the definitions that you have produced.

## Implementation in C# using Microsoft Code Contracts

From your UML model and B-Method specification, you should implement your solution to the system problem using C# and Microsoft Code Contracts. Again, you should show how the UML model and class diagrams have led to your implementation, and discuss in your report any changes from your design and specification into your final implementation.

This element of your submission should consist of your Visual Studio 2012 project submitted in an electronic form, and also a test application that will run the system, illustrating that it works correctly and all the invariants and pre-conditions in the system have been met.

Your implementation should consist of the following:

* The various components of the system implemented in C# using Microsoft Code Contracts. This means that your class files should have invariants and relevant requires on the operations.
* A test application that will allow the running of the system. How you illustrate this is up to you (user interface, random operation with text outputs, etc.). Your test application should illustrate that your application works to the defined system specification.
* Well documented and commented code. At this level, your code should have significant comments to show that you understand what you are doing, and also allow the generation of API documentation if required.

## Report

Your coursework submission must be supported by a report. **The maximum page count for the report is 20 pages. The report should have standard margins and use a 12pt font**. The report should take the following form:

* Introduction to the coursework and the system to be developed, including any assumptions that you have made about the system to be developed.
* A UML design (see the UML model submission element)
* B-Method definitions, with descriptions of any modifications and refinements made during the development process.
* Comparison of your original design against the final implementation, noting any changes made.
* Evaluation of the final implementation.
* Conclusion summarizing the above, and also providing a critical analysis of using the B-Method and Microsoft Code Contracts to develop a system in this manner.

# Marking Scheme

The coursework must be submitted to the School of Computing office, room C34 Merchiston, by 4pm on **Friday 27th of November.** Any late submissions that are not authorized by your Programme Leader will be capped at 40%. Any evidence of plagiarism will be submitted to the School misconduct officer for possible disciplinary procedure.

The assignment is worth **60%** of the total module marks. Your assignment will be marked as follows:

|  |  |
| --- | --- |
| **Item** | **Mark Available** |
| UML Model of System, including requirements analysis, static structure, and behavior analysis | 20 |
| Abstract Machine Notation specification of system using the B-Method. This should include initial specification, any refinements, and proof obligations | 40 |
| Microsoft Code Contracts implementation of the system based on the UML design and B-Method specification | 20 |
| Report, including comparison of final implementation against stages of development, evaluation of the final application, and a critical analysis of using the B-Method and Microsoft Code Contracts | 20 |
| **Total** | **100** |