The assignment is marked out of 30, on a scale according to the standard University marking scheme. If you are used to a different scale, then don’t be surprised if you receive a much lower mark than you had expected - see the description of the marking scale on the course website.

Especially if you do not have a lot of previous programming experience, you should concentrate on the tasks required for a “good” mark. Do not spend a disproportionate amount of time on the more advanced tasks - it is possible to obtain a good mark, even if these are omitted.

The assignment will be assessed on a range of criteria including completion of the various tasks, correctness, and presentation of the code and the worksheet. The following guide shows the typical requirements for each grade - however, to obtain these grades, code must also be well-structured and readable, including layout and variable names, with accurate, appropriate comments; any errors must be minor; and the worksheet text must be clear and concise with a good standard of grammar.

**Pass (diploma only): 12-14**
Submit & demonstrate a working controller with three options [1]. Demonstrate the supplied cache proxy test running in Eclipse [6].

**Good: 15-17**
Submit & demonstrate a working controller which uses a hashmap [2], a working cache proxy [5], and a JUnit test which correctly identifies the index error [7].

**Very Good: 18-20**
Submit & demonstrate a working controller which uses a hashmap and reads the subjects from the property file [3], and a working cache proxy [5]. Submit & demonstrate a JUnit test which correctly identifies the category B error [8], and explain this clearly on the worksheet.

**Distinction: 21-23**
*In addition to the above:* on the worksheet, clearly explain why the controller is hard to test and how this might be improved [13.2]. Explain why the composition of the Random and Cache proxies is difficult [13.3]. Submit & demonstrate a JUnit test which correctly identifies the category C or D error [9], and explain this clearly on the worksheet. Submit Javadoc which clearly and correctly describes the implemented classes [14].

**Exceptional: >23**
Marks above 80% are exceptional and will only be awarded where this is justified according to the criteria on the Common Marking Scheme. As a minimum, this requires faultless, professional-quality code, and well-reasoned answers to all of the worksheet questions (including [13.4]) which demonstrate a significant insight into the design problems.