The assignment is marked out of 70, 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. Attempting the more advanced tasks without a good solution to the basic tasks is likely to result in a lower mark.

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.

**Good:** 35-41

Submit & Demonstrate code and screenshots for an application that allows the user to navigate between different locations and look in different directions [8]. On the worksheet, provide a clear description of the classes used in your application. This should involve multiple classes with appropriate relationships [17.1]. Complete any additional tasks allocated at the demonstration session [24].

Small implementation bugs are acceptable, as long as the design is good, and the overall application demonstrates the basic functionality. Submissions with more serious bugs, poor class design, or poor worksheet answers may be awarded a Pass (diploma only).

**Very Good:** 42-48

Submit & Demonstrate code and screenshots for multiple “portable items” which the user can pick up and put down in different locations [12]. Submit & Demonstrate an interface which uses multiple input methods (menus, buttons, etc.) [11]. On the worksheet, provide a clear description of the classes used in your application, including both locations and portable items. This should involve multiple classes with appropriate relationships [17.1].

**Distinction:** 49-55

In addition to the above: Submit & Demonstrate application code with a well-designed class model. On the worksheet, explain this model, and provide a good comparison of some plausible alternative designs [17.2]. Submit & Demonstrate application code which is portable and provides an intuitive and robust interface.

**Exceptional:** >56

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 clear, well-reasoned answers to all of the worksheet questions. If these conditions are met, then an implementation of any advanced features [13,14,15] may also be taken into account.