FIT9131 Assignment B, S1 2022 - Where is Migaloo?

FIT9131 Assignment B: Where is Migaloo?

Introduction

This assignment is due by 11.55pm Friday of Week 12 (27th May, 2022). It is worth 35% of the marks for your final assessment in this unit. **Heavy penalties will apply for late submission.** This is an individual assignment and must be your own work. You must attribute the source of any part of your code which you have not written yourself. Please note the section on plagiarism in this document.

In preparing your program please note the following:

- **You must use the workspace environment in the Ed platform to code all parts of your program. You must not copy and paste large sections of code from somewhere else.**
- **You must acknowledge all code in your assignment that you have taken from other sources**.
- **The Java source code for this assignment must be implemented according to the Java Coding Standards for this unit.**
- **Only a text interface is to be used for this program**. **More marks will be gained for a program that is easy to follow with clear information/error messages**.

Any points needing clarification may be discussed with your tutor in your applied learning class. You should not make any assumptions about the program without consulting your tutor.

Completion of this assignment contributes towards the following FIT9131 learning outcomes:

1. Design and construct Java programs according to standard object-oriented principles

2. Apply and demonstrate debugging processes to Java applications
3. Develop strategies for efficient and effective program testing
4. Document code according to specific programming standards
5. Identify and apply the "object-oriented" concepts of encapsulation, abstraction and polymorphism
6. Explain and apply software engineering principles of maintainability, readability and modularisation

Specification

For this assignment you will simulate the work of citizen scientists who collect information about whales that visit the eastern coast of Australia each year. This section specifies the required functionality of the program.

Background

Each year during winter and spring different species of whales can be seen along the eastern coast of Australia as they travel north on their annual migration from the Southern Ocean and then travelling south as they return. The migrating whales are of great interest to tourists who observe the whales from the shore or boats, and to marine biologists who study the numbers and health of the different species of whales.

For information about the whales the tourists and marine biologists rely on observation data gathered by volunteer citizen scientists from four different locations along the eastern coast of Australia (*Eden*, *Jervis Bay*, *Byron Bay*, *Hervey Bay*). The citizen scientists are trained to identify the main species of whales that visit this part of Australia. *Humpback*, *Minke*, *Southern Right* whales are the most common, and rarely seen are *Blue* whales and *Orcas* (Killer Whales). Of particular interest in recent years has been the sighting of Migaloo, a male adult Humpback whale with an unusual white colouring.

Each day during the whale watching season the citizen scientists report details of any whale sightings at their location. For each sighting the scientists report the species of whale, its direction of travel, whether it is an adult or calf (baby whale) and if it appears to be injured.

To give tourists an idea of the most likely places to see whales, especially interesting whales (rare whales, calves, and Migaloo) the citizen scientists' observation data is used to calculate a Whale Watching Location Desirability factor (WWLD). The WWLD for each location is calculated for each location each day with the observation data for the season.

Whale Watching simulation

The Whale Watching program will simulate one day of observations.

The simulation begins with a welcome message.

The user is then prompted for the name of the citizen scientist at each observation point. You may assume there is only one citizen scientist at each observation point.

The numbers of adults and calves for each species of whale (Humpback, Minke, Southern Right, Blue, Orca) that have been observed at each location so far during the season are read from a text file **seasonObservations.txt**. The file has 4 lines, with 11 comma separated numbers on each line. Each line represents the location. Within each line the numbers of sightings of adults and calves for each species of whale are listed in order and the last number is the number of sightings of Migaloo. There is no other reading from the file during the actual *running* of the program.

The program then sets up the whales in each of the four locations for the current observations as follows:

1. At each location there is a random number of 0-9 adult whales, with each number having an equal probability.
2. The species of whale will be one of the following: Humpback, Minke, Southern Right, Blue, Orca. The percentages of each species of whale typically sighted on the eastern coast of Australia is shown in the following table:

Species of Whale	%
Humpback	50
Minke	25
Southern Right	20
Blue	3
Orca	2

3. (Hint: to calculate the probability of sighting a particular species of whale, generate a random number from 1 to 100. There is a 1% chance of each of these numbers being generated so you can nominate numbers 1-50 for a Humpback, 51-75 for a Minke, etc.).
4. There is a 1% chance that a humpback adult whale will be Migaloo, the white whale. Note that Migaloo can only be sighted at most once each day overall locations.
5. The direction of travel of a whale is north or south, with an equal probability of each.
6. If a whale is travelling south then it may have an accompanying 0-1 calves, with each number having an equal probability.
7. Each whale (adult or calf) has a 10% chance of being injured.

The simulation considers each location in turn. At each location the citizen scientist reports the whale sightings, the season observations summary is updated and the WWLD is calculated. When all locations have been visited, a summary of sightings from all locations for the day is reported, the most desirable whale watching location is reported and the updated season summary is written back to the file **seasonObservationsUpdated.txt**

Specific actions at each location

At each location the following actions are performed.

1. The program displays the following information:
    1. Total number of whales sighted (including calves)
    2. For each adult whale sighted:
        1. species of whale
        2. direction of travel (north or south)
        3. if it has an accompanying calf
        4. whether it is injured
    3. In addition, a message is displayed if Migaloo was sighted, giving the direction of his travel and if he appears injured.
2. The season observations data is updated.
3. The Whale Watching Location Desirability factor (WWLD) for the location is calculated from the season observation data, as follows:

$$WWLD = total\_adults + 2 \times total\_calves + 4 \times total\_rare\_whales + 10 \times Migaloo$$

Note: Blue whales and Orcas are considered to be rare whales.

Specific actions at the completion of observations at all locations

After the observations have been completed at all locations, the following summary is displayed on the screen.

1. For each type of whale:
    - total number observed over the four locations
    - total number of calves
    - total number of injured whales
    - whether Migaloo was sighted
2. The most desirable location to view whales, i.e. the location with the highest WWLD.

Program and Class Design

The design of the program will be discussed in the Workshop and Applied classes in week 9. It is important that you attend these classes.

Important Notes

1. Your program must demonstrate your understanding of the object-oriented concepts and general programming constructs presented in FIT9131. Consider carefully your choice of classes, how they interact and the fields and methods of each class. You must use appropriate data structures to store the various objects (location, whales, etc.) in the program. You must make use of **both Arrays and ArrayLists** in your program. Make sure that you discuss your design with your tutor. You must document any additional assumptions you made.
2. You will be required to justify your design and the choice of any data structures used at the

interview.

3. Validation of values for fields and local variables should be implemented where appropriate. You should not allow an object of a class to be set to an invalid state (i.e. put some simple validations in your mutator methods).
4. Your program should handle incorrect or invalid input and present the user with relevant error messages. No invalid input should crash the program.
5. Exception handling should be used where appropriate.

Assessment

Assessment for this assignment will be done via an **interview** with your tutor. The marks will be allocated as follows:

- 8% - Test strategy for the **Whale** class.
- 27% - Java code and object-oriented design quality. This will be assessed on appropriate implementation of classes, fields, constructors, methods and validation of the object's state.
- 10% - Progress of test strategy and code development, as shown via Ed workspace environment. Your tutor will assess your work during your applied session in weeks 10 and 11.
    - 5% in week 10 for Whale class and its test strategy
    - 5% in week 11 for at least two classes in addition to Whale class
- 55% - Program functionality in accordance to the requirements.

A reminder that you must use the workspace environment in the Ed platform to code all parts of your program. You must not copy and paste large sections of code from somewhere else and you must acknowledge all code in your assignment that you have taken from other sources.

Marks will be deducted for untidy/incomplete submissions, and non-conformances to the FIT9131 Java Coding Standards.

You must submit your work by the submission deadline on the due date (a late penalty of 10% per day, inclusive of weekends, of the possible marks will apply). There will be no extensions - so start working on it early.

All submitted source code must compile. Any submission that does not compile, as submitted, will receive a grade of 'N'.

Interview

You will be asked to demonstrate your program at an interview following the submission date. At the interview, you will be asked to explain your code/design, modify your code, and discuss your design decisions and alternatives. Marks will not be awarded for any section of code/design/functionality that you cannot explain satisfactorily (the marker may also delete excessive in-code comments before you are asked to explain that code).

In other words, you will be assessed on your understanding of the code, and not on the actual code itself.

For **on-campus students**, interview times will be arranged in the applied classes in Week 12 and will take place on campus.

For **online students**, the interviews will be organised in the applied classes in Week 12 and will take place online via zoom. You must have audio and video capabilities on your computer that can be used for the interview.

It is your responsibility to make yourself available for an interview time. **Any student who does not attend an interview will receive a mark of 0 for the assignment.**

Submission Requirements

The assignment must be submitted by **11.55 pm Friday of Week 12 (27th May 2022)**.

The submission requirements for Assignment B are as follows:

- The main class in your program **MUST** be called **WhaleWatching.java** and it should contain the **main()** method to start the program.
- Test strategy for Whale class submitted as a pdf file.
- Submit all your work (coding and test strategy) via the Ed platform.
- Re-submissions are allowed before the submission deadline. Please ensure that you do not click on submit button after the due date. The last submission will be used for grading purposes and a submission after the deadline will incur a late penalty.
- A signed Assignment Cover Sheet. [Note: You are required to download the Assignment Coversheet, sign the document and upload the pdf file in the Ed platform(you may drag and drop to the Toggle Pane)]

Marks will be deducted for any of these requirements that are not complied with.

Warning: there will be no extensions to the due date. Any late submission will incur a 10% per day penalty. It is strongly suggested that you submit the assignment well before the deadline, in case there are some unexpected complications on the day (e.g. interruptions to your home internet connection).

Plagiarism and collusion

Plagiarism and collusion are viewed as serious offences. All submitted code will be subjected to a similarity checker, and any submissions determined to be similar to a submission from a current or past student will be investigated further. The outcome of the decision pertaining to plagiarism and/or collusion will be determined by the faculty administration. If it is determined that plagiarism or collusion has occurred then you may be severely penalised, from losing all marks for the assignment, to facing disciplinary action at the Faculty level. To ensure compliance with this requirement, be sure

to do all your coding in the Ed workspace environment and do not copy and paste any code into the workspace environment.

In cases where cheating has been confirmed, students have been severely penalised, from losing all marks for an assignment, to facing disciplinary action at the Faculty level. Monash has several policies in relation to these offences and it is your responsibility to acquaint yourself with these.

Student Academic Integrity Procedure (https://www.monash.edu/__data/assets/pdf_file/0004/2300935/Student-Academic-Integrity-Procedure.pdf))