

CS1083 Assignment #1 – Winter 2021

Due: Friday, January 22nd before 4:00pm (Atlantic), submitted in the Assignment 1 dropbox in Desire2Learn. (Read the submission instructions at the end of this document carefully).

The purpose of this assignment is to:

- Review classes and objects and partially filled arrays.

Important habits to develop for all assignments:

- Start each assignment early so you have time to ask questions if / when you encounter problems.
- When working on the computer, save your work early & often.
- Include **Javadoc** for each program that you write (for drivers only your name and student number beside the @author tag in a Javadoc comment block at the top of the driver class is required). Run the Javadoc utility on your classes to ensure there are no errors in your comments (the documentation files created from running the Javadoc utility only need to be submitted with the assignment when specified):
`javadoc -author -private FileName.java`
- Don't forget to make a back-up copy of your work somewhere other than on your computer (e.g.: on a USB key, external hard drive, or cloud drive).

This assignment is to be done individually. If you have questions, direct them to a tutor/assistant during a help session in the "Faculty of Computer Science Student Success Centre" team or to your course instructor.

Loading a Ferry

You are writing software to manage the booking system for small ferries which carry a small number of vehicles and the passengers in the vehicles (the ferries do not allow pedestrians). The drivers of the vehicle must pre-book their crossing in advance. The booking system must adhere to restrictions on the total number of vehicles and total number of axels allowed on a ferry, and weight limits. Each ferry also has a restriction on the number of passengers allowed on the ferry. You will need to write two classes, one called Vehicle and one called Ferry.

The Vehicle class contains instance data for:

- A unique ID for the vehicle (automatically assigned)
- The number of passengers in the vehicle (the driver is included in this count)
- The number of axels (this is the number of pairs of wheels i.e.: a car has 2 axels, a car towing a trailer has 3 axels, transport trucks can have even more axels i.e.: 3, 4, 5 etc.)
- Weight of the vehicle (in kg)

Each of the instance data requires an accessor. A mutator is needed to set the number of passengers in a vehicle to a new value. Each vehicle should also be automatically assigned a unique identification number when it is created. These ID numbers should be assigned in order, starting at 100.

The Ferry class manages the vehicles it holds. The following information is passed in to the constructor:

- The maximum number of vehicles allowed on the ferry.
- The maximum number of axels allowed on the ferry.
- The maximum number of passengers allowed on the ferry.
- The weight limit of the ferry (in kg).

This class needs to accomplish the following:

1. Add a vehicle to the ferry, returning true or false to indicate if the vehicle is successfully added. A vehicle can be added to the ferry provided that it doesn't exceed the maximum number of vehicles, and only if all of the following additional conditions are met: a) the vehicle to be added would not put the ferry over its weight limit; b) the vehicle to be added would not put the ferry over its passenger limit; c) the vehicle to be added would not put the ferry over its axel limit; and d) the vehicle is not already on the ferry. Once a vehicle is added the total weight of all vehicles, total number of passengers, and total number of axels must be updated.
2. Remove a vehicle from the ferry, returning true or false to indicate if the vehicle has been removed. A vehicle can only be removed if it is already on the ferry. Once a vehicle is removed, the total weight of all vehicles, total number of passengers, and total number of axels must be updated.
3. Change the number of passengers in a vehicle that is already on the ferry, returning true or false to indicate if the number of passengers has been updated. The number of passengers in a vehicle can only be changed if a) the vehicle is already on the ferry and b) the change in the number of passengers would not put the ferry over its passenger limit. If the number of passengers in a vehicle is updated, the total number of passengers is also updated.
4. Report the weight that is still available on the ferry (weight limit – total vehicle weight).

5. Retrieve a textual vehicle list, formatted in tab-separated columns, showing the ID and weight of the vehicle. At the end of the list print the total weight, total number of axels, and total number of passengers, as shown in this example:

```
100      01,524.75 kg
101      30,480.50 kg
102      27,130.00 kg
Total Weight:      59,135.25 kg
Total Axels:      12
Total Passengers:  9
```

Notes:

- The order of the vehicles in the vehicle list is not important.
- You must use arrays (and not ArrayLists) to store vehicles in the Ferry class.
- Vehicles are identified by their ID when searching through the list of vehicles.
- You may find it helpful to create a helper method in the Ferry class to return the position of a vehicle in the list.

Driver program for testing:

Write a driver program to test the Vehicle and Ferry classes, covering the following:

- Add a small number of vehicles.
- Try to add a vehicle that is already booked on the ferry.
- Try to add a vehicle that would put the ferry over its weight limit.
- Try to add a vehicle that would put the ferry over its passenger limit.
- Try to add a vehicle that would put the ferry over its axel limit.
- Change the number of passengers of a vehicle on the list, for cases where the new number of passengers would and would not put the ferry over its passenger limit, and for the case where the vehicle doesn't exist.
- Try to remove a vehicle that is on the ferry.
- Try to remove a vehicle that is not on the ferry.
- Report the remaining weight available on the ferry.
- Print a vehicle list.

The output of your driver program should indicate the purpose of each test case and print the effects on the vehicle list and total weight, axels, and passengers after each successful operation.

Submission instructions are on the next page...

Your electronic submission (submitted via Desire2Learn) will consist of two files:

1. a written report. This should begin with a title page that includes: the course (CS 1083), your section (FR01B, FR02B, FR03B), the assignment number, your full name, and your UNB student number. That should be followed by two sections, with each part clearly identified with a section heading. Include:
 - a. all the source code for your solution (with Javadoc comments included),
 - b. the sample output from running your application.

This written report should be prepared using a word processor; we recommend using Microsoft Word (i.e. create a .docx file for your report). Copy & paste your java source code & required output into the report document. Add appropriate headings for each part. Fix up the formatting where necessary, adjusting line breaks & page breaks to ensure that your document is easy to read. Use a monospaced font for your code to maintain proper indentation.) Once the report is complete and you've checked it all over, save the .docx file for your own records, and then **save a second copy in pdf format for submission.** (Note: Be sure to open that file in a pdf viewer to verify that the pdf was generated correctly.) **The SINGLE pdf file containing your report will be submitted to the appropriate assignment drop box on Desire2Learn.** (It is important that you submit a pdf file and NOT the original Word document. This pdf will allow the marker to write comments directly on your work to give you better feedback.)

Note: name this report as follows: **YourName_CS1083_As1_Report.pdf**

2. an archive file (.zip) that contains your Java source code for this assignment. Make sure that your archive includes all .java files (in case the marker wishes to compile & run your code to test it). You should not include the report document or the .class files in your archive. This archive should be submitted as a **single file** to the appropriate drop box on Desire2Learn.

Note: name this archive file as follows: **YourName_CS1083_As1_Archive.zip**