

Module: CMP-4008Y Programming 1
Assignment: Coursework 1 - Electronic Component Inventory Program

Set by: Dr Gavin Cawley (g.cawley@uea.ac.uk)
Date set: 1st November 2019
Value: 20%

Date due: 5th December 2019 3pm
Returned by: 13th January 2020
Submission: evision

Learning outcomes

The aim of this assignment is for the student to gain experience in the design and implementation of a relatively small program using an object oriented approach. This will further reinforce the student's grasp of fundamental concepts such as classes, objects, instance variables and methods, but will also introduce the use of multiple classes, each of which define a different type of object. During the assignment, the student will also begin to gain object-oriented design skills by deciding how the state and behaviors of the objects should be implemented and how the objects should interact.

On successful completion of this exercise, the student will have reinforced a basic understanding of the concepts of classes and objects, will be familiar with the basic syntax of Java programming constructs used to implement classes, instance variables and methods and will be able to select the instance variables and methods required to represent the state and behaviors of simple types of real-world objects. The student will have gained familiarity with some of the classes in the Java standard libraries used for file based I/O and collections. The student will also have an improved ability to use the ideas of cohesion and coupling to guide the design of programs comprised of more than a single class definition.

Specification

Overview

Chartlins is a company that specialises in providing electronic components for hobbyists to build simple electronic circuits of their own design, or from a kit of parts. They have hired you to write a system to manage their inventory of components etc. The program must be written in the Java programming language and demonstrate the use of classes and object-oriented design to write modular, maintainable programs. The file `inventory.txt` provides details of the company's inventory of electronic components (resistors, capacitors, diodes, transistors and integrated circuits). For each stock item, the file provides the component type, the stock code, the number of items in stock and the unit price in pence. The remainder of the record provides some additional information:

- Resistors — the resistance in Ohms (see en.wikipedia.org/wiki/RKM_code for details).
- Capacitors — the capacitance in Farads, encoded in a manner similar to resistance.
- Transistors — an indication whether they are NPN, PNP or FET devices.
- Integrated Circuits (ICs) — a brief description.

The program must read the data in this file into memory, stored in an appropriate set of objects, and print answers to the following queries on the console:

- Print a list of the inventory, sorted in order of increasing price.
- What is the component with the largest number of components in stock?
- How many NPN transistors does Chartlins have in stock?
- What is the total resistance of all of the resistors in stock?
- How many stock items have unit prices above 10p?

The program should consist of three files, which define the basis of the marking scheme:

- `StockItem.java`, which should define a class that records the information about a stock item.
- `Inventory.java` which should define a class representing the inventory of the company, using some appropriate data structure or container. The class should have methods that assist in answering the same queries as before (the methods should be generic so they could potentially be used to answer similar queries, rather than specifically those given in the specification).
- `StockProgram.java` This file contains the `main` part of the program that loads and processes the inventory data, and provides answers to the five queries.

Note that `inventory.txt` may change before the due date for the assignment, and the answers to the queries must be based on the final version of that file.

Relationship to formative assessment

This assignment builds on the skills gained from the laboratory exercises for cmp-4008Y that have already been completed, for which formative feedback can be obtained from the teaching assistants during your scheduled laboratory session.

Deliverables

Your solution must be formatted using the PASS program, available on all laboratory machines, to produce a PDF file containing the source code of the program, the compiler messages (if any) and the output of the program. The PDF must then be submitted via evision. *Your solution will only be marked if it has been submitted via evision.*

The PASS program is not able to provide input to your program via the keyboard, so programs with a menu system, or which expect user input of some kind are not compatible with PASS. Design your program to operate correctly without user input from the keyboard.

We strongly recommend that the student also uploads this .pdf file (or alternatively a .zip file containing the source code for your program) to BlackBoard. There may be some issues that are not easily resolved using only the printed code listing, and the markers may wish to compile and run the program for themselves. This may affect the mark awarded either positively (by reassuring the marker that the code executes correctly and efficiently) or negatively (for instance because the code does not execute correctly or is inefficient).

If you develop your solution on a computer other than the laboratory machines, make sure that you leave time to test it properly with PASS, in case of any unforeseen portability issues. If your program works correctly under IntelliJ on the lab machines, but does not operate correctly using PASS, then it is likely that the data file (`inventory.txt`) you are using has become corrupted in some way. PASS downloads a fresh version of the file to test your submission, so this is perhaps the most likely explanation.

Resources

- <https://stackoverflow.com/> - An excellent site for finding information about specific issues relating to various programming languages, including Java. It is important however not to become too reliant on sites such as StackOverflow, which are great for details, but don't give the "big picture", so it is difficult to get a good understanding of programming in this way. Note that if you re-use or modify code found on-line, then you must provide a comment giving the URL and a brief explanation of what modifications have been made. Re-using code found on-line without proper attribution would constitute plagiarism.
- <https://docs.oracle.com/javase/tutorial/> - A set of tutorials, provided by Oracle, who now develop the Java programming language.
- <https://docs.oracle.com/javase/8/docs/api/> - Java Application Programming Interface (API) documents. Very useful for finding out what methods a class from the Java standard libraries provide.

Plagiarism and collusion: Stackoverflow and other similar forums (such as the module discussion board) are valuable resources for learning and completing formative work. However, for assessed work such as this assignment, you may not post questions relating to coursework solutions in such forums. Using code from other sources/written by others without acknowledgement is plagiarism, which is not allowed (General Regulation 18)

Marking Scheme

Marks will be awarded according to the proportion of specifications successfully implemented, programming style (indentation, good choice of identifiers, commenting etc.), and appropriate use of programming constructs. It is not sufficient that the program generates the correct output, professional programmers are required to produce maintainable code that is easy to understand, easy to debug when (rather than if) bug reports are received and easy to extend. The code needs to be modular, where each module has a well-defined interface to the rest of the program. The function of modules should be made as generic as possible, so that it not only solves the problem specified today, but can easily be modified or extended to implement future requirements without undue cost. The code should also be reasonable efficient. Marks may also be awarded for correct use of more advanced programming constructs not covered in the lectures.

The marking scheme is as follows (out of 100 marks):

- StockItem.java - 40 marks.
- Inventory.java - 25 marks.
- StockProgram.java - 10 marks.
- The remaining 25 marks are awarded for the answers to the five inventory questions listed above (five marks each).