Description

Your task is to create a program which simulates an online shop selling computers with different characteristics (e.g. CPU, speed, hard drive, graphics card, make). The program should be able to match customers looking for a computer with specific needs (e.g. speed greater than 2.1 Ghz, etc.) with computers that are in stock at the shop.

All interaction with the program should be done with a text based menu (non-graphical menu) which gives the user the choice of selecting various commands, such as create a new computer to be placed in stock, create a new customer looking for a computer, match the needs of a customer with computers in stock, etc.

The details for the implementation of the system are given in the steps below:

1. Design and implement an abstract class `Computer` describing a general purpose computer. (2 marks).

2. Design and implement the concrete (non-abstract) classes `Laptop` and `Desktop` which are subclasses of class `Computer`. The two classes should contain all the relevant methods and fields found specifically in laptop and desktop computers respectively. (5 marks)

3. Design and implement an abstract class `Person` describing the characteristics and operations of a human. (2 marks).

4. Create a class `Customer` inheriting from class `Person` which describes a person who would like to purchase a computer. Such a customer is in need for a particular type of computer (laptop or desktop) with particular characteristics (e.g. a laptop with an upper limit in weight, battery duration, speed, etc. or a desktop with a lower limit in monitor size, hard drive capacity, etc.) (5 marks).

5. Create a text based menu (non graphical menu) which gives the user the choice of selecting various commands described below:

   (a) Option 'c' creates a new computer (laptop or desktop) with specific characteristics entered by the user: cpu type, cpu speed, hard disk capacity, graphics card type, battery (for laptop only), weight (for laptop only), price, etc. This option should give the user two sub-options 'l' and 'd' for the creation of a laptop or desktop computer respectively. (15 marks)
(b) Option ‘n’ creates a new customer looking for a computer. This option should ask the user for the specific characteristics of a computer that this person is looking for. For example, price limit, desktop or laptop, cpu speed, hard disk capacity, upper limit of weight (for a laptop only), minimum battery life (for a laptop only), graphics card type, etc. (11 marks)

(c) Option ’s’ (search) gives a user the possibility to search for a particular type of a computer already existing in the system, based on price, cpu speed and other characteristics. (15 marks)

(d) Option ’m’ (match) matches all customers in the system with a single computer in the system based on the needs of a customer and the computer characteristics. Assume that in the case of limited resources (not enough computers with appropriate characteristics) a customer A who was created in the system before customer B has priority over B assuming that they both match a specific computer. (15 marks)

(e) Option ’o’ (order or otherwise sort computers) displays all existing computers according to price in descending order, i.e. the most expensive computer should be displayed first of characteristics, the second most expensive computer should be displayed next, etc. (10 marks)

(f) Option ’l’ (list or otherwise sort customers) displays all existing customers according to requirements of a computer with CPU speed in ascending order, i.e. the customer who requires the slowest computer will be displayed first, the customer requiring the second slowest computer will be displayed second, etc. (10 marks)

**Marking Scheme:** The marks achieved for each part of the program are indicated in the description of the task above. In addition to these the following will be taken into account:

- *Code readability* (structure, comments, variable naming, etc.): 5%
- *Implementation* (e.g. quality, efficiency, etc.): 5%

Programs which do not conform to the design specifications of the task (e.g. the abstract and concrete classes described are not implemented) will be imposed a penalty of 10%.

The maximum for work which does not compile is 25%.

**Submission of assignments using a different method other than Blackboard will not be accepted and zero (0) marks will be awarded in such cases.**

**Deadline:** Monday 14th December 2009, 5:00pm.

**Submission Instructions**

*Files to submit:* all your source code files (the .java files NOT the .class files) in plain ASCII format. This should include test classes, i.e. classes which were implemented for the sole purpose of testing other classes.

You should submit via BlackBoard’s Assignment functionality (do NOT use email, as email submissions will be ignored.), all all the files described above. A single zip file containing all the above files could be submitted alternatively.
Note that Blackboard will allow to make a submission ONLY ONCE. So make sure before submitting (i.e. before pressing the Submit button), that all the files you want to submit are contained there (or in the zip file you submit). Submissions with missed files cannot be resubmitted, so do NOT email me with such requests.

The following describes how to submit your work via BlackBoard:

1. Access http://learning.wmin.ac.uk and login using your username and password (if either of those is not known to you, ask the HelpDesk at the Library.).

2. Click on the module's name, MODULE: 2AIT515.2009 OBJECT-ORIENTED SOFTWARE DEVELOPMENT found under My Modules & Courses.

3. Click on the Assignments button found on the left hand side menu.

4. Click on View/Complete Assignment.

5. Attach your files containing in your submission, by using the Browse button repeatedly (A single zip file containing all your files can be submitted alternatively.). Note that all the names of your files must have an extension, otherwise you will not be able to upload them.

6. Fill in the requested information:

   • Comments: Type your full name and your registration number, followed by:

     "I confirm that I understand what plagiarism is and have read and understood the section on Assessment Offences in the Essential Information for Students. The work that I have submitted is entirely my own. Any work from other authors is duly referenced and acknowledged."

7. Click the Submit button.

If Blackboard is unavailable before the deadline you must email me before the deadline with a copy of the assignment, following the naming, title and comments conventions as given above and stating the time that you tried to access Blackboard. You are still expected to submit your assignment via Blackboard. Please keep checking Blackboard’s availability at regular intervals up to and after the deadline for submission. You must submit your coursework through Blackboard as soon as you can after Blackboard becomes available again even if you have also emailed the coursework to me.