COM1003 Java Programming
Autumn Semester 2020-21
Programming Assignment 1

Dr Siobhán North
Department of Computer Science, The University of Sheffield

Learning outcomes

This assignment will assess your ability to:

- Write a program from a specification;
- Write clear, good quality program code;
- Use numbers and strings in Java;
- Use the `sheffield` package.

This assignment is worth 12% of your mark for the first semester of the module and must be submitted by Monday 26th October. You will find information about the exact deadline, the marking scheme and how you must submit your work at the end of this document.

You do not need to do everything described below to hand in your work and get marks for it. The marks will be awarded depending on how much you have achieved (see below for details) but if you hand in a program it must compile and run to get any marks. So a well written program that does something is always better than a program which would have done more if it had worked.

The specification below is quite precise in telling you what sort of input you should expect and what you must output but usually does not tell you how to get from one to the other. This is deliberate and part of the test. However it is not meant to be ambiguous so if there is something you don't understand you can ask questions, ideally by email. I will put all the email questions and their answers on an FAQ page as they come in. Even if you think you understand everything it is a good idea to check this page before you submit.

In the days before COVID 19 it was possible to pay for shopping in cash. This assignment relates to that lost world.

The First Task

Write a program called `Change.java` that asks the user how much their shopping cost, how much cash they offered to pay for it and tells the user what their change should be.

An example of what could appear in the command window is given below

```
        How much is your shopping? 12.34
        How much cash did you offer? 20
        Your change should be 7.66
```

Please note name of the program and the format and wording of the output and input is part of the specification and you will lose marks if you don't follow it exactly including the spelling, capitalization and punctuation. Do not add your own headings or blank lines, just stick to the specification above because your work will be checked by computer. For full marks the change you output should be a real number expressed to two decimal places. You are required to use the `sheffield` package for both input and output in this assignment and no marks are available for using anything else.

You can assume that

- The user will type in a positive number less than 1000 for both the price and the money offered and nothing else
- The price and cash offered could be either an integer or a real number to two decimal places but nothing else
- The price of the shopping will be less than or equal to the money offered

If you get this far with a perfect program you can expect to get 30%. Once you have got this to work save a copy of your program in case of accidents before modifying it to attempt the rest of the assignment.

The Second Task

Extend your program to calculate and display how the change should be made up. So the output, should you do this task as well, could look like this.

```
        How much is your shopping? 123.45
        How much cash did you offer? 140
        Your change should be 16.55
        Made up of
        10 pound notes:  1
        5 pound notes:   1
        2 pound coins:   0
        1 pound coins:   1
        50 pence coins:  1
        20 pence coins:  0
        10 pence coins:  0
        5 pence coins:   1
        2 pence coins:   0
        1 pence coins:   0
```

This is quite repetitive so the bulk of the marks will be for getting the first 2 lines right. After that do as many of the subsequent lines as you have time for but do them in the order shown.

The assumptions mentioned above still hold but you can also assume that the cash offered will not be inappropriate given that £20 is the largest note in common circulations so the cash offered will not be at most £19.99 more than the price.

Hint: Work out what the change is in pence and use that value for the calculations.

If you get this far with a perfect program you can expect to get 60%. Once you have got this to work save a copy of your program in case of accidents before modifying it to attempt the rest of the assignment.

The Third Task

Extend your program to ask the user for the name of a file. Your program should then open the file, which will contain six real numbers, one to a line, to two decimal places. They represent the price and amount offered for three items of shopping in the order price, cash offered, price, cash.... Your program should read them in from the file and display them in a table with a header followed by three lines, one for each price, cash pair but the output lines should contain three values; the price, the cash and the change. The output should be arranged neatly in right justified columns.

So if you had a file called "input.txt" which contained the values 12.34, 15.00, 34.56, 35.00, 56.78, 60.00 each on a line by itself with no commas then the output would end with a table like this.

```
Price     Paid  Change
12.34    15.00    2.66
34.56    35.00    0.44
56.78    60.00    3.22
```

This should follow the output from the tasks above so the whole thing is shown overleaf.

You have been provided with the file `input.txt` to practice with. Save it in the same directory as your program. Your program will be tested with a file with a different name containing six different values but all the assumptions mentioned above will still hold, so the amount offered will be greater than or equal to the price and the numbers will be between 0.00 and 999.99

If you get this far with a perfect program you can expect to get 100%.

The third task is independent of the second task so, if you can't do that task, you can still do the 3rd task but if you do both the 2nd and the 3rd you must do them in that order.

```
How much is your shopping? 123.45
How much cash did you offer? 140
Your change should be 16.55
Made up of
10 pound notes:  1
5 pound notes:   1
2 pound coins:   0
1 pound coins:   1
50 pence coins:  1
20 pence coins:  0
10 pence coins:  0
5 pence coins:   1
2 pence coins:   0
1 pence coins:   0
What is the name of your file? input.txt
 Price    Paid  Change
12.34   15.00    2.66
34.56   35.00    0.44
56.78   60.00    3.22
```

Submission and deadline

You must submit a file called Change.java via the submission point on Blackboard by 3pm on Monday 26th October. Do not submit anything else and do not submit it in any other format. Blackboard is very bad at displaying Java programs so you may think that Blackboard has messed up the layout of your program but I will download the program text and the layout will be preserved.

Late work will be penalised using the standard University scale (a penalty of 5% per working day late; work will be awarded a mark of zero if it is more than 5 working days late).

This is an individual assignment. You must work on it alone and hand in your own work. If you work collaboratively and then pretend you did the work alone we will find out (we have a very good plagiarism checker and all submitted work will go through it) and, as you have already been told or are about to be told during your first group tutorial, we take the use of unfair means in the assignment process very seriously. Don't even think about handing in work you didn't do yourself.

The Marking Scheme

You will get two, equally weighted, marks for this assignment, one for the working version and one for programming style.

The Working Program

Task	1	2(start)	2(last 7 lines)	3
Use of EasyReader	9%			17%
Use of EasyWriter	10%	4%		21%
Calculation	2%	10%	7%	
Meets specification*	9%	2%	7%	2%
Total	30%	16%	14%	40%

* The mark for following the specification means the specification as far as the program is completed so incomplete programs will not be penalized under this heading.

Program Style

The Programming Style mark will be calculated as follows and scaled to the maximum for whatever version you submit.

Correct use of variables	25
Correct use of types	10
Correct use of constants	15
Indention and layout	25
Comments	25