Assignment on the first part of the course

Elementary Programming – 02318

Preamble:

This assignment will count for 20% of your final grade.

The topics we will cover with this assignment can be found on the **SemesterPlan.pdf** that is available both on DTU Inside and Microsoft Teams.

Topics go from lesson held on September 3rd 2020 to lesson held September 30th 2020.

Any questions can be addressed to me (emia@dtu.dk) or our TAs.

Deadline:

The deadline is at 8AM on October 28th 2020.

The Problem:

Italy is not as advanced as Denmark. Our customer is a mom and pop store in Giulianova in the Abruzzi region in Central Italy.

They sell fruit and vegetables and they want a system to apply a discount to what they sell. They want also to know how many invoices with discount they are selling.

They sell three types of fruit and vegetables:

- Cucumbers
- Bananas
- Apples

They will read a file called **invoice.txt** with the right information for the program to calculate the discount and the final price. This task is mandatory.
The program will update a file called **stats.txt** where the customer will find number of invoice and how many invoice with discount they have. This is an extra task.

Tasks:

- **[Mandatory]** The program must read the information from invoice.txt and then calculate the discount and final price, which is to be printed to console (printf) in the manner shown in this assignment.
- **[Extra Credit]** The program should update a file called stats.txt with a running tally of the amount of invoices the customer has in total, as well a show many invoices with discount they have.
-

The discount applied is as following:

- If we sell more than 5 cucumbers we will apply a 5% discount
- If we sell more than 9 bananas we will apply a 10% discount
- If we sell between 5 and 9 bananas we will apply a 5% discount
- No discount is applied to apples

The price of fruit and vegetables in given in the file because they change the price, usually weekly depending on how much they pay to their fruit and vegetable resellers.

The program will need to validate the file as well as to see if the format expected is correct or not.

The program will accept a file in this format:

```
GROCERY    QUANTITY    UNITPRICE
1          12          1.4
2          25          1.5
3          65          6.0
```

Grocery is the type of fruit or vegetables:

- Cucumber is 1
- Bananas is 2
- Apples is 3

Quantity is the number of items per type, in this example we have:

- 12 Cucumbers
- 25 Bananas
- 65 Apples

The Unit Price is the price for a single item.
Without applying the discount, the cost would be the following:

- 12 Cucumbers will cost EUR 124.80
- 25 Bananas will cost EUR 262.50
- 65 Apples will cost EUR 390.00

With discount applied:

- 12 Cucumbers will cost EUR 118.56 (5% discount applied = EUR 6.24)
- 25 Bananas will cost EUR 236.25 (10% discount applied = 26.25)
- 65 Apples will cost EUR 390.00 (no discount applied)

The program will print the following output:

```
 GROCERYTYPE     PRICE    DISCOUNT APPLIED
           1    118.56               6.24
           2    236.25              26.25
           3    390.00               0.00
```

Other scenarios for calculating discount and validation:

Scenario No Discount

Given this input file:

```
GROCERY    QUANTITY    UNITPRICE
1          1           10.4
2          1           10.5
3          1           6.0
```

This will be the console output:

```
GROCERYTYPE     PRICE    DISCOUNT APPLIED
           1     10.40               0.00
           2     10.50               0.00
           3      6.00               0.00
```

Scenario Discount Bananas between 5 (not included) and 9

Given this input file:

```
GROCERY    QUANTITY    UNITPRICE
1          1           10.4
2          6           10.5
3          1           6.0
```

This will be the console output:

```
GROCERYTYPE     PRICE    DISCOUNT APPLIED
           1     10.40               0.00
           2     59.85               3.15
           3      6.00               0.00
```

Scenario Bad Header

Given this input file:

```
GROCERY    QUANTITY
1          12          10.4
2          25          10.5
3          65          6.0
```

This will be the console output:

```
ERROR Header is wrong
```

Scenario Bad Type

Given this input file:

```
GROCERY    QUANTITY    UNITPRICE
1          1           10.4
4          1           10.5
3          1           6.0
```

This will be the console output:

```
GROCERYTYPE     PRICE    DISCOUNT APPLIED
          1     10.40                0.00
ERROR: found bad grocery type: 4
```

Other scenarios for keeping **stats.txt** updated:

The initial file format is the following (you will find it in the content of the assignments) inside **stats.txt**:

```
INVOICES:    0
WITHDISCOUNT:    0
```

After executing three consecutive times the file from scenario “Scenario No Discount” this will be the updated situation inside **stats.txt**:

```
INVOICES:    3
WITHDISCOUNT:    0
```

After executing three consecutive times the file from scenario “Scenario Discount Bananas between 5 (not included) and 9” this will be the updated situation inside **stats.txt**:

```
INVOICES:    6
WITHDISCOUNT:    3
```

Nothing will be printed on console, the file **stats.txt** will be updated.

How is the evaluation going to work?

The grading on this assignment will constitute 20% of your total grade on this course.
While completing the mandatory task is enough to earn a full grade on this assignment, the optional task will give you a chance of a full grade in case mistakes have been made in the mandatory task.
You need to do the tasks by yourself so the implementation of the evaluation doesn’t change that.
You will be evaluated on two things:

- all the scenarios need to work correctly as listed on this document
- you need to have all the right checks in places, like for example always closing a file or check there are no

errors before using a specific assignment (like when you use fopen you need to check for NULL)

Advices on implementing the tasks

Everything that you have to do here, has for sure something to do with what we studied. Remember that all code done in class is available at https://github.com/invasionofsmallcubes/elementary-programming-dtu and so you can see some examples.
Copy and paste is not enough because you will need to make it work to pass the scenarios.