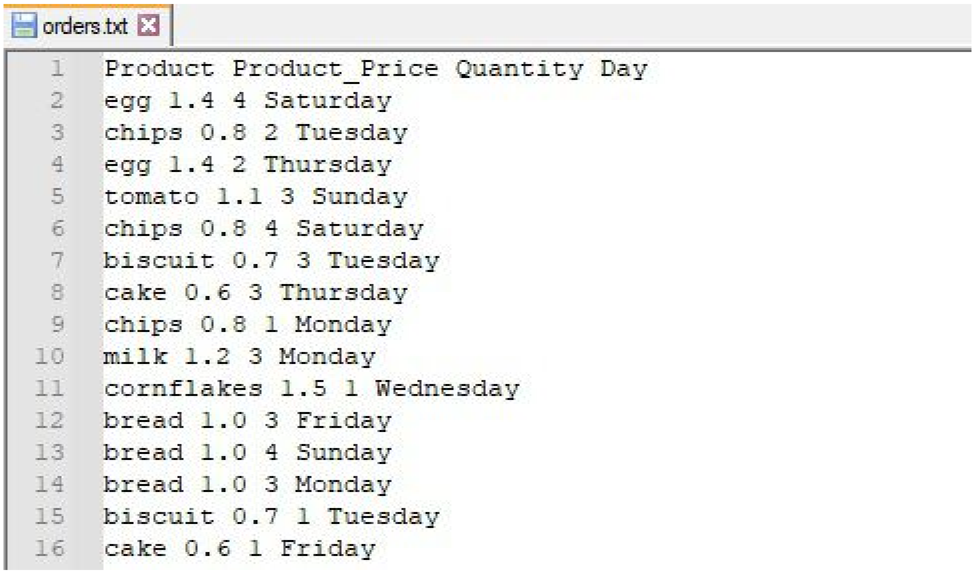
ITU Computer Engineering Department
BLG 223E Data Structures, Fall 2021
Recitation #6

Problem Definition

You have an orders.txt file whose name will be given as a **command line argument** in this homework. Each row indicates an order in this text file. There are 4 columns which are seperated by one space. For each order, the first column indicates the product, the second column indicates the product price, the third column indicates the product quantity and the fourth column indicates the order day. In this recitation, mainly you calculate each production total revenue from orders. 5 products with the highest revenue will be displayed on the screen. Revenue is calculated for each order as below:

$$revenue = product\ price \times quantity$$



```
Product Product_Price Quantity Day
egg 1.4 4 Saturday
chips 0.8 2 Tuesday
egg 1.4 2 Thursday
tomato 1.1 3 Sunday
chips 0.8 4 Saturday
biscuit 0.7 3 Tuesday
cake 0.6 3 Thursday
chips 0.8 1 Monday
milk 1.2 3 Monday
cornflakes 1.5 1 Wednesday
bread 1.0 3 Friday
bread 1.0 4 Sunday
bread 1.0 3 Monday
biscuit 0.7 1 Tuesday
cake 0.6 1 Friday
```

Figure 1: 'orders.txt' file.

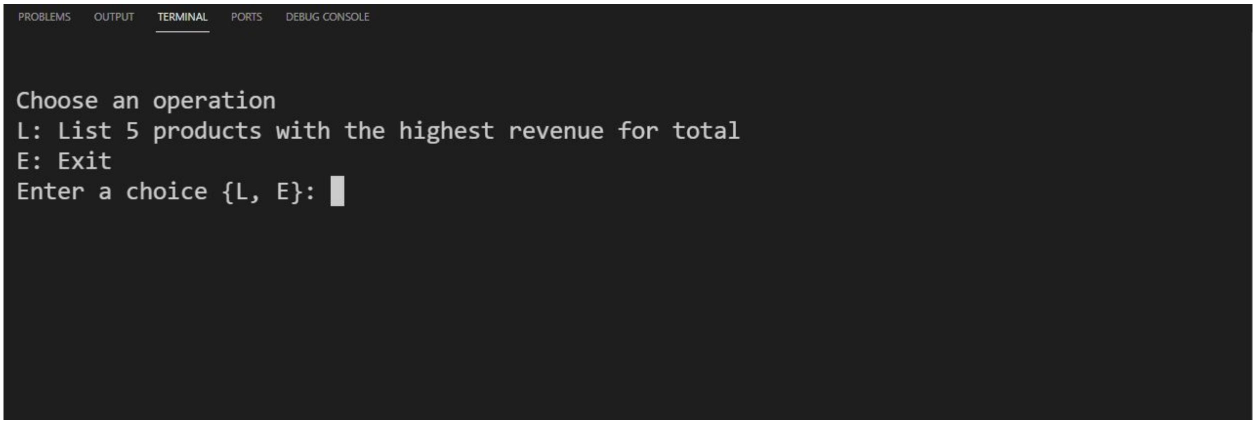
Revenue for the first order in 'orders.txt' file is calculated as below:

$$\#1\ order\ revenue = 1.4 \times 4 = 5.6$$

fstream library can be used in order to read from orders.txt file. You can create a map value which holds the total revenue for each product. So, its key and value types should be <string,double>. When you read file row by row and column by column from orders.txt file, you can calculate total revenue for each product using this map value. Afterwards, Multimap, map iterator and multimap reverse_iterator can be used to sort the total products revenue.

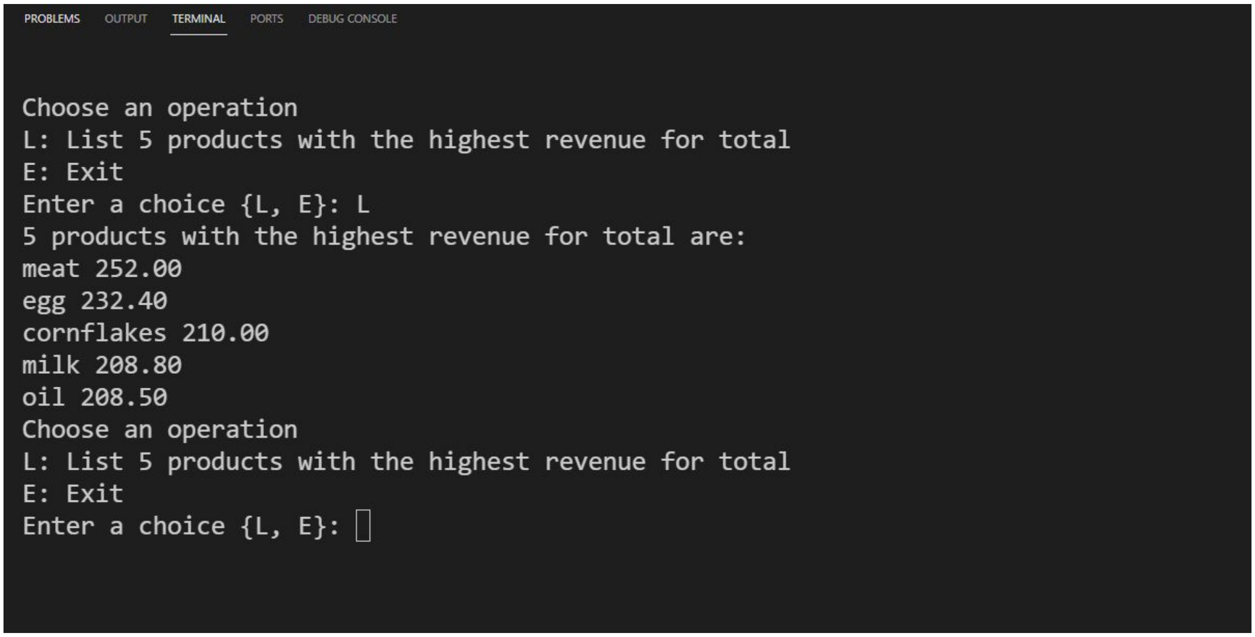
You should only fill listProducts function in the code file. There is a sample scenario below.

Example



```
Choose an operation
L: List 5 products with the highest revenue for total
E: Exit
Enter a choice {L, E}: 
```

Figure 2: Terminal screen when the program is run.



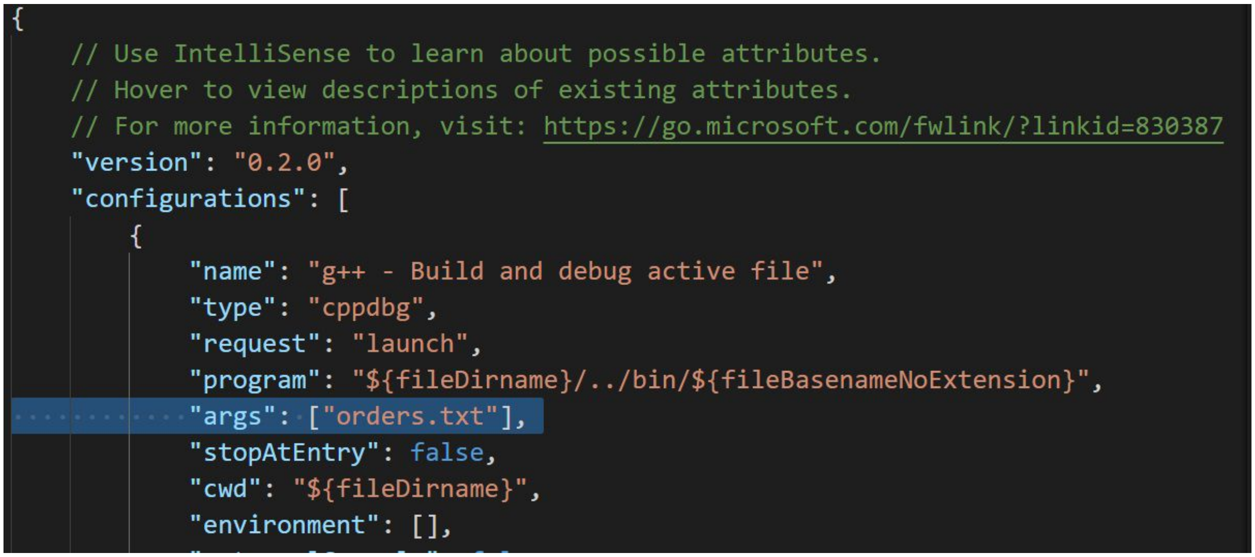
```
Choose an operation
L: List 5 products with the highest revenue for total
E: Exit
Enter a choice {L, E}: L
5 products with the highest revenue for total are:
meat 252.00
egg 232.40
cornflakes 210.00
milk 208.80
oil 208.50
Choose an operation
L: List 5 products with the highest revenue for total
E: Exit
Enter a choice {L, E}: 
```

Figure 3: When you select ’L’, the output of the program.

Note: You can use **’setprecision’** function from **’iomanip’** for two decimal precision display. You can find the explanations about ’setprecision’ function at the link below.

Link: https://www.cplusplus.com/reference/iomanip/setprecision/

Note: To get the file path(name) from the command line argument, the args parameter of the launch.json file should be changed as in the figure below.



```
{
    // Use IntelliSense to learn about possible attributes.
    // Hover to view descriptions of existing attributes.
    // For more information, visit: https://go.microsoft.com/fwlink/?linkid=830387
    "version": "0.2.0",
    "configurations": [
        {
            "name": "g++ - Build and debug active file",
            "type": "cppdbg",
            "request": "launch",
            "program": "${fileDirname}/../bin/${fileBasenameNoExtension}",
            "args": ["orders.txt"],
            "stopAtEntry": false,
            "cwd": "${fileDirname}",
            "environment": [],
```

Figure 4: The args parameter of the launch.json file.

Submission Rules

- Make sure you write your name and number in all of the files of your project, in the following format:

  /* @Author
  Student Name: <student_name>
  Student ID : <student_id>
  Date: <date> */

- Use comments wherever necessary in your code to explain what you did.

- Your program will be checked by using **Calico(https://bitbucket.org/uyar/calico)** automatic checker.

- Do not share any code or text that can be submitted as a part of an assignment (discussing ideas is okay).

- Only electronic submissions through Ninova will be accepted no later than deadline.

- You may discuss the problems at an abstract level with your classmates, but you should not **share or copy code** from your classmates or from the Internet. You should submit your **own, individual** homework.

- Academic dishonesty, including cheating, plagiarism, and direct copying, is unacceptable.

- A different input file will be used for evaluating the homework.

- If you have any question about the recitation, you cand send e-mail to Yunus Emre Cebeci(cebeci16@itu.edu.tr).

- Note that **YOUR CODES WILL BE CHECKED WITH THE PLAGIARISM TOOLS!**