Credit Card II Project, Fall 2025

Tong Yi

Warning:

1. This is copyrighted materials; you are not allowed to upload to the Internet.
2. Our project is different from similar products in Internet.
   (a) Ask help only from teaching staff of this course.
   (b) Use solutions from artificial intelligence like ChatGPT or online tutoring websites like, but not limited to, chegg.com, violates academic integrity and is not allowed.

1 Task A: Find the maximum monthly total across all categories; draw a chart for a selected category

1. Create a file named `monthly_category_total_chart.cpp`. The input file is a yearly credit card report containing up to 12 months and no more than 20 categories.
2. In the program, prompt the user to enter the name of an input file.
3. Read the file.
4. Identify and sort all categories alphabetically.
5. Find the maximum monthly total across ***all*** categories.
6. Display all categories alphabetically and prompt the user to choose one.
7. Display a chart of asterisks representing the monthly totals of the selected category.

1.1 Example of Task A

Create a file named `spending.csv` with the following contents. On Mac/Linux, you can use VS Code, TextEdit, Vim, or Emacs. On Windows, you can use VS Code, Notepad, Notepad++, or Vim. The file `spending.csv` should contain only one column of double-precision numbers.

```
Date,Description,Category,Amount
01/16/2024,Con Edison,Utilities,91.35
02/14/2024,National Grid,Utilities,32.75
03/17/2024,Macy's,Shopping,109.3
03/19/2024,Taxi,Travel,71.37
```

```
03/24/2024,Marshalls,Shopping,98.76
03/30/2024,National Grid,Utilities,10.93
03/31/2024,MTA,Travel,20.38
05/25/2024,Macy's,Shopping,32.87
06/15/2024,Macy's,Shopping,55.49
08/21/2024,Banana Republic,Shopping,59.85
08/25/2024,National Grid,Utilities,27.16
```

The file has three categories, listed in alphabetical order.

```
Shopping
Travel
Utilities
```

Here is a monthly total for each category based on the above data.

Month	Shopping	Travel	Utilities
1	0	0	91.35
2	0	0	32.75
3	109.3 + 98.76 = 208.06	71.37 + 20.38 = 91.75	10.93
4	0	0	0
5	32.87	0	0
6	55.49	0	0
7	0	0	0
8	59.85	0	27.16
9	0	0	0
10	0	0	0
11	0	0	0
12	0	0	0

Explanation:

1. On 3/17/2024, we spent 109.30 on shopping, and on 3/24/2024, another 98.76 on shopping. Therefore, the total spent on shopping in March 2024 is 109.30 + 98.76 = 208.06.
2. Similarly, on 3/19/2024, we spent 71.37 on travel and on 3/31/2024, another 20.38 on travel. Therefore, the total spent on travel in March 2024 is 71.37 + 20.38 = 91.75.

March 2024 Spending Summary

Shopping

Date	Amount
03/17/2024	109.30
03/24/2024	98.76
Total	208.06

Travel

Date	Amount
03/19/2024	71.37
03/31/2024	20.38
Total	91.75

Utilities

Date	Amount
03/30/2024	10.93
Total	10.93

Here is a sample output for the above `spending.csv`.

```
Enter a file name: spending.csv (with return key)
select one of the following categories
0.Shopping
```

```
1.Travel
2.Utilities
choose a number in [0, 2]: 0 (with return key)
max monthly total across all categories = 208.06
MONTH  Shopping TOTAL
Jan              0.00
Feb              0.00
Mar            208.06 ****************************************
Apr              0.00
May             32.87 ******
Jun             55.49 **********
Jul              0.00
Aug             59.85 ***********
Sep              0.00
Oct              0.00
Nov              0.00
Dec              0.00
```

another sample input/output:

```
Enter a file name: spending.csv (with return key)
select one of the following categories
0.Shopping
1.Travel
2.Utilities
choose a number in [0, 2]: 1 (with return key)
max monthly total across all categories = 208.06
MONTH    Travel TOTAL
Jan              0.00
Feb              0.00
Mar             91.75 *****************
Apr              0.00
May              0.00
Jun              0.00
Jul              0.00
Aug              0.00
Sep              0.00
Oct              0.00
Nov              0.00
Dec              0.00
```

yet another sample input/output:

```
Enter a file name: spending.csv (with return key)
select one of the following categories
0.Shopping
1.Travel
2.Utilities
```

```
choose a number in [0, 2]: 2 (with return key)
max monthly total across all categories = 208.06
Jan             91.35 ****************
Feb             32.75 ******
Mar             10.93 **
Apr              0.00
May              0.00
Jun              0.00
Jul              0.00
Aug             27.16 *****
Sep              0.00
Oct              0.00
Nov              0.00
Dec              0.00
```

1.2 Number of Asterisks

1. `MAX_NUM_ASTS` is assumed to be 40.
2. The number of asterisks representing the maximum monthly total across ***all*** categories is `MAX_NUM_ASTS`.
3. In the above `spending.csv`, the maximum monthly total across all categories is 208.06 (as seen in Shopping in March) and is represented by 40 asterisks.
4. For any other monthly total, the number of asterisks is calculated as

   `(monthly_total / max_monthly_total_across_all_categories) * MAX_NUM_ASTS`.

   The decimal part should be truncated.

   Example: In the above `spending.csv`, the maximum monthly total across all category is 208.06. In May 2024, the total spending in Shopping is 32.87. The number of asterisks is calculated by (32.87 / 208.06) * 40 = 6.31. After **truncating** decimals, this results in 6 asterisks in Shopping in May 2024.

1.3 Submission of Task A

Submit `monthly_category_total_chart.cpp` to gradescope. Note that the grading script uses random double numbers to test. As a result, your output will be different in each running.