| | |
|---|---|
| **Name:** | **QUID:** |

**Midterm Instructions:** The exam is 120 minutes long. Answer all first five questions, question 6 is bonus. Use proper indentation/alignment in your code, the lines of the answer sheets are formatted for this purpose.

**Q.1** **[6 POINTS]** Every food delivery company has several drivers, that varies from one delivery company to another. Each driver delivers several orders in a day, that also varies from one driver to another. Each order has a different bill amount based on the order. Once again, the bill amount varies from one order to another. Write a program that acquires the number of drivers, then for each driver it requires the number of delivered orders then for each order it requires the bill amount. The program should:

a. For each driver, display the total amount for all bills of orders delivered by a driver.
b. For all drivers, display the total amount for all bills of orders for all drives altogether.
c. Display the driver number that has the minimum total amount of bills of delivered orders.

**Q.2** **[6 POINTS]** Consider the file **donations.txt** below, the file has many more records than what is showing here. Each line represents a donation record having the information: code of the donation event, date and time of donation, donor name, donation amount, and donor's rating about the satisfaction of the event out of 10.
**Note that some donations do not have donor's rating.**
Write a program to:

a. Display **total amounts of donations** for the event with the code HDYYW00E013 by Unknown donors.
b. Display **the number of donations** by Unknown donors in October regardless of the year.
c. Display the **codes of donations** that have been made between 10:30 **am** and 5:30 **pm** with a rating higher than 6.
d. Write all **unique events' donation codes** to the file **events.txt**.

Your program should handle the exception "FileNotFoundError".

```
Code,Date yyyy-mm-dd time hh:mm:ss,Donor’s Name,mount$,Rating
TT7373WY12,2023-10-01 08:15:00,Unknown,100000$,7
TT7373WY12,2023-10-03 17:20:00,Khaled,5000$
DAIF6826Y6,2023-03-14 22:25:00,Sarah,600$,8
DAIF838H53,2023-05-29 22:25:00,Unknown,8000$
…
```

**Q.3** **[6 POINTS]** Write the full code of a program that reads the total bill amount in QR and a string that represents the code of discount voucher. The program validates the code of the discount voucher by calling the function validateVoucher(code) that returns the discount percentage if the voucher is valid, otherwise it returns 0. The program displays the amount to be paid after deducting the discount from the total bill amount.

The code of a discount voucher is valid if it satisfies all the following conditions: (1) code length is twelve characters, (2) first three characters are letters, (3) the first and third letters are capital while the second letter is small case, (3) the characters from the fourth to the tenth have only one uppercase letter, one underscore, two lowercase letter, and three digits in any order (4) the last two characters are digits representing the discount percentage.

**Q.4** **[6 POINTS]** Write a function called highestRated(ratingList) that receives a two-dimensional list containing the product name, store name, and rating of the product. The function returns a dictionary mapping a product name with a tuple containing information about the store having the highest rating of that product including: the store and rating.

**Example:** for the input ratingList: [["USB Switch 8", "e-max", 6] , ["MS Surface 9", "virgin", 8] , ["Lenovo 24", "Amazon" , 9], ["MS Surface 9", "Alibaba" , 7], ["Lenovo 24", "Alibaba", 8] ]

The function returns the dictionary: {'USB Switch 8': (6, 'e-max'), 'MS Surface 9': (8, 'virgin'), 'Lenovo 24': (9, 'Amazon')}.

**Q.5** **[6 POINTS]** Write the full code of the function magicSquare(intlist) that receives a **squared** 2-D list of **integers** and returns True if it is magic square 2-D matrix otherwise, it returns False.

A 2-D matrix is a magic square matrix if using the numbers from 1 to the size of the matrix **only once**, the sum of the sum of any row equals the sum of any column and equals the sum of any diagonal in the matrix.

**Example:** below are squared 2D lists inputs and the corresponding results returned by this function.

$$\begin{bmatrix} 2 & 7 & 6 \\ 9 & 5 & 1 \\ 4 & 3 & 8 \end{bmatrix}, \quad \begin{bmatrix} 2 & 16 & 13 & 3 \\ 11 & 5 & 8 & 10 \\ 7 & 9 & 12 & 6 \\ 14 & 4 & 1 & 15 \end{bmatrix}, \quad \begin{bmatrix} 9 & 8 & 6 \\ 5 & 4 & 7 \\ 1 & 3 & 2 \end{bmatrix}, \quad \begin{bmatrix} 10 & 5 & 8 & 9 \\ 13 & 7 & 3 & 1 \\ 4 & 15 & 11 & 2 \\ 4 & 13 & 6 & 14 \end{bmatrix}$$

True True False False

**Q.5** **[2 POINTS]** Write the full code of the class Student that holds the following attributes about a student: name, quid, major, and grades list. Code the following:

- Class Student having the constructor __init__ that receives name, quid, major, and grades list of the student. The class also has the method __str__ to return a string representing the object state (the values of its attributes).
- A program that
  - creates a list of students of Student class having the following values for their attributes:

| name | quid | major | grades |
|---|---|---|---|
| Huda | 200206364 | CS | [90,87,79,95,68] |
| Iman | 200104812 | EE | [93,91,88,97] |
| Sarah | 200200130 | CE | [75,73,68,81,83,77] |

  - Display the data for each student on the screen:
  - Create a function that receives a list of Student objects and returns a dictionary where quid is the key, and the grades average is the value.