United International University
Department of Computer Science and Engineering

DS 1502: Programming for Data Science Laboratory

Final Examination: Fall 2024

Total Marks: 25 Time: 1 hour 15 minutes Set: A

Answer all 2 questions. Numbers to the right of the questions denote their marks.

Any examinee found adopting unfair means will be expelled from the trimester / program as per UIU disciplinary rules.

1. You are tasked with processing a text file containing information about a list of employees in a company. The file is named `employees.txt` and follows this format: `(Employee ID, Name, Department, Salary)`. An example is given below: [12]

```
101, Alice, HR, 50000
102, Bob, IT, 60000
103, Charlie, Marketing, 55000
104, Diana, IT, 65000
105, Eve, HR, 52000
```

Perform the following tasks:

(a) Read the content of this file and store it in an appropriate data structure.

(b) Create a list of names that contains exactly 2 vowels in their names from the stored data. Print the list.

(c) Calculate the average salary of all the employees from the stored data and print it.

(d) Calculate the average salary for each department (the average of all the salaries of all the employees of the department). Create a new dictionary that has the departments as keys and the average salary of each department as values. Print the dictionary.

Example: To test your code, create an `employees.txt` file with the data as mentioned above and then the expected outputs would be:

- List of names containing exactly 2 vowels: `['Eve']`
- Average Salary of All Employees: 56400.0
- The dictionary of the average salaries of all the departments: `{'HR': 51000.0, 'IT': 62500.0, 'Marketing': 55000.0}`

2. You are given a list of dictionaries representing product information:

```
products = [
    {"id": 1, "name": "Laptop", "price": 1200, "category": "Electronics"},
    {"id": 2, "name": "Smartphone", "price": 800, "category": "Electronics"},
    {"id": 3, "name": "Headphones", "price": 150, "category": "Electronics"},
    {"id": 4, "name": "Sofa", "price": 700, "category": "Furniture"},
    {"id": 5, "name": "Table", "price": 300, "category": "Furniture"}
]
```

Write a Python program with the following functions: [3 + 2 + 4 + 4 = 13]

(a) **findExpensiveProducts(products, threshold)**: This function takes the list of product dictionaries and a price threshold as arguments. It returns a **list of names** of products whose price is greater than or equal to the threshold.

(b) **getAveragePrice(products)**: This function takes the list of product dictionaries and returns the **average price** of all products.

(c) **groupProductsByCategory(products)**: This function takes the list of product dictionaries and returns a **dictionary** where the keys are categories and the values are **lists of product names** belonging to that category.

(d) **getUniquePrices(products)**: This function takes the list of product dictionaries and returns a **set** of unique price values.

Finally, test your functions with the following inputs and print the results:

```
threshold = 500
print("Expensive Products:", findExpensiveProducts(products, threshold))
print("Average Price:", getAveragePrice(products))
print("Products Grouped by Category:", groupProductsByCategory(products))
print("Unique Prices:", getUniquePrices(products))
```

Expected Output:

```
Expensive Products: [‘Laptop’, ‘Smartphone’, ‘Sofa’]
Average Price: 630.0
Products Grouped by Category: {‘Electronics’: [‘Laptop’, ‘Smartphone’, ‘Headphones’],
                               ‘Furniture’: [’Sofa’, ’Table’]}
Unique Prices: {150, 300, 700, 800, 1200}
```