Karadeniz Technical University – Artificial Intelligence and Data Science Department
Programming - II – Term Project I
Start Date: 24/04/2025 13:00
End Date: 30/05/2025 17:00

Lecturer: Asst. Prof. Çağatay Murat Yılmaz

Detailed information will be given in the lecture.

Assignments must be uploaded from https://uek.ktu.edu.tr/ogrenci.html.

Projects must be done by groups of two or three students.

Turkiye Neighborhood Information Management and Analysis System

The neighborhoods.txt file (taken from https://www.e-icisleri.gov.tr/Anasayfa/MulkiIdariBolumleri.aspx) contains the names of the neighborhoods (mahalle) and the province (il) / district (ilçe) / town (belde) or province center (il merkezi) to which they are connected, respectively. For example,

> The row " ÜNİVERSİTE TRABZON -> ORTAHİSAR -> ORTAHİSAR-DISTRICT CENTER " refers to the ÜNİVERSİTE neighborhood in ORTAHİSAR district center of TRABZON province.

> The row "ESENTEPE ADIYAMAN -> ADIYAMAN-PROVINCE CENTER" refers to the ESENTEPE neighborhood in the center of ADIYAMAN province.

> The row "ATATÜRK AFYONKARAHİSAR -> ÇOBANLAR -> KOCAÖZ" refers to ATATÜRK neighborhood in KOCAÖZ town of ÇOBANLAR district of AFYONKARAHİSAR province.

Develop a program that performs the following operations in Python programming language.

Rules:

- ✓ Operations on data should be done with the help of lists.
- ✓ Error handling must be used in parts that are likely to generate errors (not finding a file, not being able to read, etc.).
- ✓ All operations should be done by loading a module into the main program. All different functions should be done in separate functions.
- ✓ The program should be written following the rules of PEP 8 - Style Guide for Python Code [https://peps.python.org/pep-0008/].

When the program is first run, it should present a menu that performs the following operations. The user should be able to perform operations through this menu.

Desired Functions

1. Searching

1.1. Search for exact matching neighborhood (user will enter only neighborhood name and exact neighborhood name matching will be searched). As a result of the listing, province, district, and neighborhood name will be given. If there is no match, the user must be warned.

1.2. Search for a neighborhood in a partial match (user will enter only the neighborhood name and a partial neighborhood name match will be made). Province, district, and neighborhood name will be given as a result of the listing. The match can be from the beginning, middle or end. If there is no match, the user must be warned.

2. Listing

2.1. List all neighborhoods in a province (user will enter the province name and all neighborhoods in that province will be listed). According to user choice, the listing can be in ascending or descending order by neighborhood name (the program will offer this option). The result of the listing will be the province, district, and neighborhood name.

2.2. List all neighborhoods in a district (user will enter province/district name and all neighborhoods in that district will be listed). According to user choice, the listing can be in ascending or descending order by neighborhood name (the program will offer this choice). The result of the listing will be the name of the province, district, and neighborhood.

3. Adding

3.1. Add a new neighborhood to a province/district (user will enter the name of the province/district and neighborhood, and the insertion will be made accordingly). If there is a neighborhood with the same name, a warning will be given, and the program will continue with a new add request. The program will list and display the neighborhoods in that province/district after the add process.

4. Deleting

4.1 Delete a neighborhood in a province/district (the user will enter the name of the province/district and neighborhood, and the deletion will be made accordingly). If there is no neighborhood with the input name, a warning will be given, and the program will continue with a new delete request. After the operation, the program will list and show the neighborhoods in that province/district.

5. Updating

5.1. Update the name of a neighborhood in a province/district (the user will enter the province/district and neighborhood name, and the update will be done accordingly). If there is no neighborhood with the entered name, a warning will be given, and the program will continue with a new update request. After the operation, the program will list and show the neighborhoods in that province/district.

6. Moving Records

6.1 Move a neighborhood in a province/district to another province/district (user will enter the province/district and neighborhood name, that neighborhood will be moved to another province/district entered by the user). Related delete and add operations will be done automatically. If there is no neighborhood with the entered name, a warning will be given, and the program will continue with a new move request. The program will list and show the neighborhoods in that province/district after the operation.

7. Plotting Graphics Plotting will be done with Matplotlib.

7.1. Plot the number of neighborhoods of districts in a province as a bar graph (the user will enter the name of the province and the graph will be drawn accordingly). Graph title, axes and labels must be complete.

7.2. Draw the number of neighborhoods of the districts in a province as a pie chart (user will enter the name of the province, drawing will be done accordingly). The graph should have the ratios of the number of neighborhoods of the districts. Graph title, axes and labels must be complete.

8. Analyzing

8.1. Find the neighborhood (or neighborhoods, if any) with the most common name

8.2. Find the provinces with the least and most neighborhoods

8.3. Calculate the average number of neighborhoods at the provincial level, including standard deviation

<div align="center">

Some Example Usages

</div>

1.1. Search for a neighborhood with an exact match. For example, if the user searches for "KABASAKAL", the provinces/districts with an exact match should be listed as follows.

KABASAKAL ADANA -> ALADAĞ -> ALADAĞ-İLÇE MERKEZİ

KABASAKAL ADANA -> ÇUKUROVA -> ÇUKUROVA-İLÇE MERKEZİ

KABASAKAL DİYARBAKIR -> SUR -> SUR

KABASAKAL KAHRAMANMARAŞ -> DULKADİROĞLU -> DULKADİROĞLU-İLÇE MERKEZİ

KABASAKAL TRABZON -> ŞALPAZARI -> ŞALPAZARI-İLÇE MERKEZİ

1.2. Search for a neighborhood with a partial match. For example, if the user searches for the neighborhood "KABASAKAL", where len(neighborhood name)> 3, the provinces/districts with a partial match should be listed as follows.

KABASAKAL ADANA -> ALADAĞ -> ALADAĞ-İLÇE MERKEZİ

KABASAKAL ADANA -> ÇUKUROVA -> ÇUKUROVA-İLÇE MERKEZİ

ESKİKABASAKAL ADANA -> KOZAN -> KOZAN-İLÇE MERKEZİ

POSTKABASAKAL ADANA -> KOZAN -> KOZAN-İLÇE MERKEZİ

KABASAKAL DİYARBAKIR -> SUR -> SUR

KABASAKAL KAHRAMANMARAŞ -> DULKADİROĞLU -> DULKADİROĞLU-İLÇE MERKEZİ

KABASAKALLI MERSİN -> SİLİFKE -> SİLİFKE-İLÇE MERKEZİ

KABASAKAL TRABZON -> ŞALPAZARI -> ŞALPAZARI-İLÇE MERKEZİ

2.2. For example, if the user enters "İstanbul" as the province and "Beyoğlu" as the district, all neighborhoods in this district should be listed as follows.

YENİ MAHALLE GİRESUN -> BULANCAK -> AYDINDERE

GÜNEY MAHALLE GİRESUN -> BULANCAK -> AYDINDERE

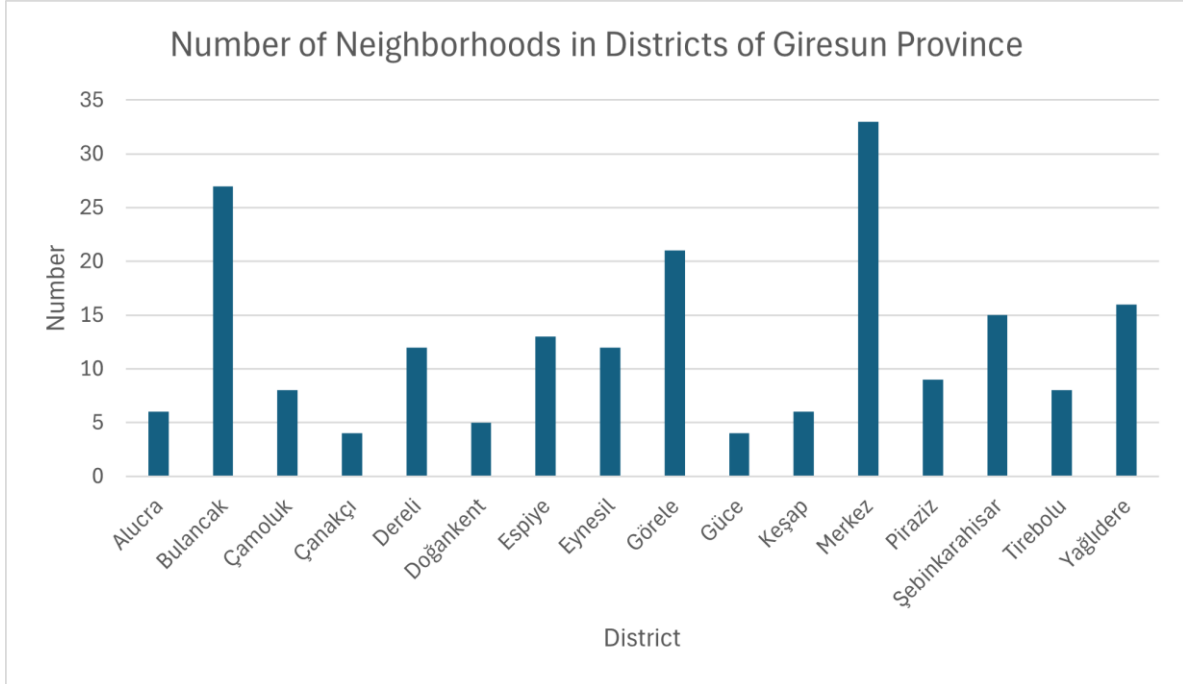*AYDINLAR GİRESUN -> BULANCAK -> AYDINDERE*

...

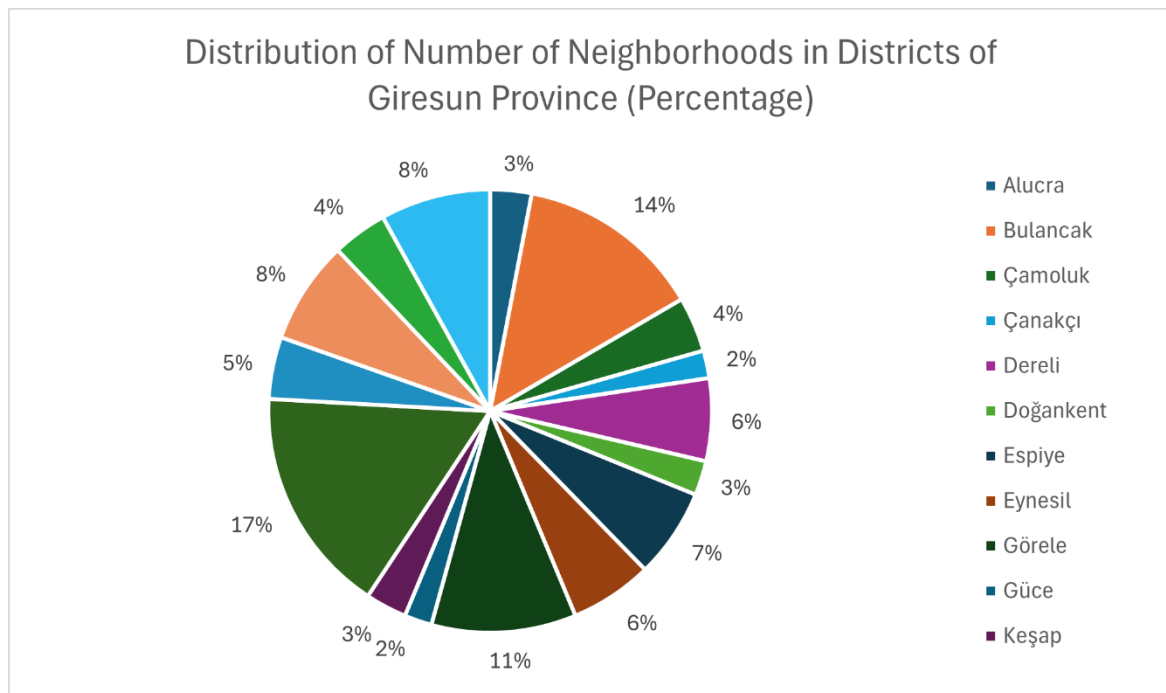*ALİBEY GİRESUN -> BULANCAK -> KOVANLIK*

GÜZELYURT GİRESUN -> BULANCAK -> KOVANLIK

SOFULU GİRESUN -> BULANCAK -> KOVANLIK

7.1. A sample bar chart of Giresun province will be as follows.

3

Number of Neighborhoods in Districts of Giresun Province

7.2. A sample pie chart for Giresun province will be as follows.

Distribution of Number of Neighborhoods in Districts of Giresun Province (Percentage)

End of project description