DATA STRUCTURES - NATIONAL PARK PROJECT

QUESTION 1

The names of the 48 National Parks in Turkey, the provinces where they are located, the dates they were declared as national parks and their surface areas (hectares) are given in the links

https://tr.wikipedia.org/wiki/T%C3%BCrkiye%27deki_mill%C3%AE_parklar_listesi

https://www.tarimorman.gov.tr/DKMP/Belgeler/Korunan%20Alanlar%20Listesi/MP-WEB-Son.pdf

a) Create a National park class that contains the following fields (you determine the appropriate data types / data structures):

NationalPark class (nationalParkName, provinceNames, announcementYear, surfaceArea)

You can use Generic List to keep the names of the provinces where a National Park is located. You only need to keep the year when the National Park was declared.

b) Create a 2-element Generic List **array** (Figure 1). Let each element of the array contain a **Generic List** of elements of the National Park class type. Taking the data sequentially from one of the links above, first create the object for the 1st National Park; then add it to the end of the list in element 0 of the array if its area is less than 15,000 hectares, and to the end of the list in element 1 if it is larger. Then add the remaining National Park objects to the appropriate list according to the specified criteria.

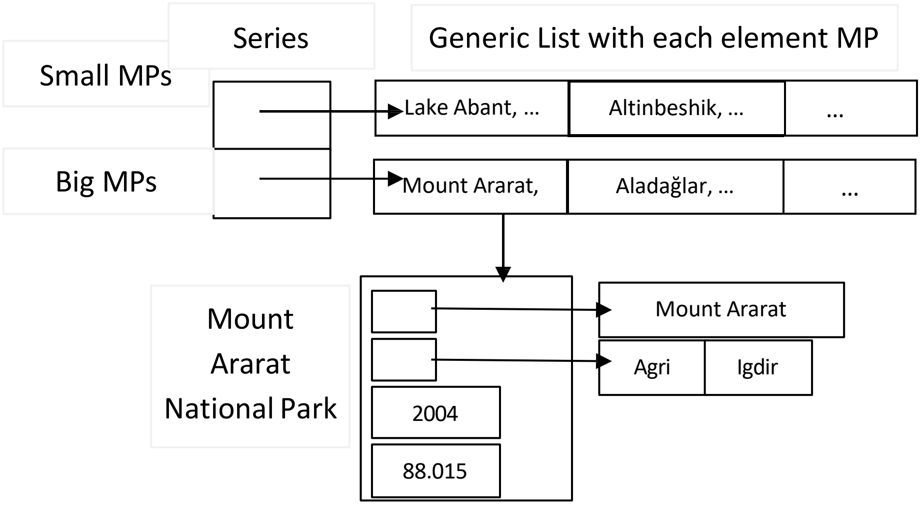

Figure 1: Array of Generic Lists (Composite Data Structure)

c) Write a method that prints all the information in each list in the array on the screen. Also calculate and print the sum of the surface area of the elements in each list.

QUESTION 2

a) Update the code or write your own to create a **Stack** National Park (class objects). Add all the National Parks in question 1 to the Stack you created (you can pull the information from the file or from the Lists you created, it's up to you). Remove all the elements from the Stack and print the National Park information on the screen.

b) Repeat the process in 2a. for **Queue** (class) Data Structure.

QUESTION 3

Design and write a **Priority Queue (class)** in C# or Java with **ascending** order and O(1) insertion time (but slower to delete the highest priority element) (the list will not be kept in order, the element will be inserted last, and the element deletion method will search for **the National Park with the smallest surface area** and delete it). You should use **List** ready data structure to hold elements of type **National Park class**. You should write all necessary methods (enqueue, dequeue, isEmpty, constructor) and their contents of the **Priority Queue** Class. Test it on the data in Question 1, print the National Parks by deleting them in priority order according to their surface area (from smallest to largest).

Note: People who make it as a console application can change the background of the Console screens to white and the text color to black:

```
Console.BackgroundColor = ConsoleColor.White;
Console.Clear();
Console.ForegroundColor = ConsoleColor.Black;
Console.WriteLine("Hello");
```