**PROJE 3**

**ARAMA AĞAÇLARI, YIĞINLAR ve HASH TABLOSU (2024)**

1. **a)** **EgeDeniziB\_Ağacı (Proje 2’deki EgeDeniziB nesnelerini içeren) ağacının oluşturulması:** Veriler **https://balikturleri.com/ege-baliklari-listesi/** bağlantısındaki sırada alınarak **Balık\_Adına** göre bir **Balıklar** adlı ikili arama ağacına eklenecektir (Şekil 1). EgeDeniziB nesnesinde balığın adı ve **Kelimeler ikili arama ağacı** bulunacaktır. Bir balığın **Kelimeler** ikili arama ağacı, ilgili balığın **Bilgi** sahasındaki her bir **kelimenin** ayrıştırılmasının ardından alfabetik sırada yeni bir ikili arama ağacına eklenmesi ile elde edilecektir. **(10)**

**EgeDeniziB Nesnesi**

**Kelimeler   
Ağacı**

………………..

…

**Şekil 1: Balıklar Ağacı**

1. Balıklar Ağacındaki ve düğümlerin içindeki kelimeler ağaçlarındaki tüm bilgileri balık adına göre alfabetik sırayla ekrana listeleyen metodu yazınız. **(10)**
2. Ortalama Kelime Ağacı Derinliği: Tüm kelimeler ağaçlarının derinliklerini buldurunuz, balık sayısına böldürünüz. Bu metoda ağacın düğüm sayısını da bulduran ve dengeli ağaç olsaydı derinliğinin kaç olması gerektiğini hesaplayan kodu da ekleyiniz. **(10)**
3. Baş harfleri klavyeden girilen iki harf arasındaki tüm Balık isimlerini listeleyen metodu yazınız (Örnek D ve H harfleri verilirse baş harfleri d ve h arasında olan balık adlarını listelemeli – Dikenli Vatoz, Eşkina, Horozbina Balığı). **(5)**
4. Tüm ağaçtaki EgeDeniziB nesnelerini balık adına göre bir diziye sıralı olarak ekleyiniz. Ardından ortadaki elemanı yeni ağacın kökü yapınız. Sonra dizinin solda kalan yarısındaki elemanların ortasındaki ve sağda kalan yarısındaki elemanların ortasındaki elemanları ağaca ekleyerek ve bu işlemi özyineli olarak tekrarlayarak dengeli bir ağaç oluşturunuz. **(5)**
5. **a)** 1. Maddede oluşturduğunuz EgeDeniziB nesnelerini **Balık Adı’na göre** bir **Hash Table’a** yerleştiren kodu yazınız [Dildeki hazır Hash Table veya Dictionary kullanabilirsiniz]. **(5)**

**b)** **Adı verilen Balığın bilgisini yine klavyeden verilecek yeni bir paragrafa göre yeniden kelime ağacını oluşturarakHash Tablosunda** güncelleyen kodu yazınız. **(5)**

**3) a)** Ders kitabı Bölüm 12’yi okuyunuz. Heap Veri Yapısı (sınıfı) ile metotlarını öğreniniz. **C# veya Java ile** **Kodlayıp çalıştırınız.** Altyapıda elemanları tutmak için dizi veya List / Vector kullanabilirsiniz. **(5)**  **[Kurşunkalemle algoritma yazma işlemi önceki senelerde kaldırılmıştır]**

**b)** Adlarına göre Balıkları **Max.** **Heap’e** (Java’daki PriorityQueue Heap düzenindedir) yerleştiren kodu yazınız (Alfabetik olarak adı sonra olan balık, ad sahasına göre daha büyüktür). **(5)** [Dileyen a maddesinde yazdığı kodu güncelleyebilir, dileyen Java’daki PriorityQueue sınıfını kullanabilir.]

**c)** İlk 3 balığı **Max Heap’ten çekerek** tüm bilgilerini listeleyen kodu yazınız. **(5)**

**4)** Ders kitabındaki 3. Bölümden (**Simple Sorting**) **veya** 7. Bölümden (**Advanced Sorting**) bir sıralama algoritması seçerek okuyunuz, öğreniniz, mantığını anlayınız.

**a)** **Kodlayarak**, Debug içerisinde değişkenlerin değişimini izleyiniz. (5)

**b)** **Zaman karmaşıklığını hesaplayınız**. Diğer bir sıralama algoritması ile karşılaştırınız, o algoritmaya göre hangi durumlarda avantajlı, hangi durumlarda dezavantajlı olduğunu kısaca açıklayınız. (5)

**c) Her iki programın, oluşturacağınız 100 elemanlı sıralanmamış bir diziyi 10.000.000 kere baştan sıralaması için geçen zamanı hesaplayınız. Zamanı ölçen kod içerisine girdi çıktı komutları yazmayınız.** (10)

**5) a)** [**https://www.youtube.com/watch?v=kPRA0W1kECg**](https://www.youtube.com/watch?v=kPRA0W1kECg)adresindeki sıralama algoritmaları görselleştirme videosunu izleyiniz. [**https://log2base2.com/**](https://log2base2.com/) **adresinden (veya benzeri bir siteden) ücretsiz 2 kısa video izleyiniz.** Görselleştirmenin, algoritmaları ve veri yapılarını anlamanıza etkisini belirtiniz. (5)

**b)** Veri Yapılarını öğrenme açısından **Videolar** ile **Etkileşimli Görselleştirme Araçlarını (Duyurularda Bağlantıları var) karşılaştırınız. (5)**

**6) Özdeğerlendirme tablosunu doldurarak raporun sonuna ekleyiniz (10 puan):**

**Özdeğerlendirme Tablosu**

|  |  |  |  |
| --- | --- | --- | --- |
| **Proje 3 Maddeleri** | **Not** | **Tahmini Not** | **Açıklama** |
| **1 a) Ağaç (EgeDeniziB İkili Arama Ağacı)** | **10** |  |  |
| **1 b) Derinlik Bulma, Ağacı Listeleme, Düğüm Sayısı Buldurma, Dengeli Ağaç Derinliği Hesaplama** | **10** |  |  |
| **1 c) Arama ve Listeleme** | **5** |  |  |
| **1 d) Kelime Ağacı Oluşturarak Kelimeleri Sayma** | **5** |  |  |
| **2) Hash Tablosu** | **10** |  |  |
| **3) Yığın Ağacı (Heap)** | **15** |  |  |
| **4) Sıralama Algoritmaları** | **20** |  |  |
| **5.a) Görselleştirmenin DSA öğrenmeye etkisi** | **5** |  |  |
| **5.b) Video & Görselleştirme Araçları Karşılaştırma** | **5** |  |  |
| **6) Özdeğerlendirme Tablosu** | **10** |  |  |
| **7) İşbölümü (hangi öğrenci hangi maddeleri yaptı ve kaç saat çalıştı?)** | **5** |  |  |
| **TOPLAM** | **100** |  |  |

Not: Projeyi tek kişi yapabileceğiniz gibi 2 veya 3 kişi ortak yapıp teslim edebilirsiniz.

Proje 3 için işletim ve kod anlatım videosu çekilmeyecektir.

Bazı soruların kodlamasında C#, bazılarında Java’yı karışık olarak kullanabilirsiniz.