

KOCAELİ SAĞLIK VE TEKNOLOJİ ÜNİVERSİTESİ

2023-2024 BAHAR DÖNEMİ

BLM324 YAZILIM LAB II

PROJE 2

Proje 2 Bilgilendirmesi

- Proje dosyalarının **son teslim tarihi 07.04.2024 saat 23.55'tir**. Bu tarihten sonra Google Classroom'daki proje teslimi kapanacaktır ve geç gönderilen projeler teslim edilemeyecektir.
- Proje ile ilgili sorularınızı mail üzerinden Arş. Gör. Eray Dursun'a iletebilirsiniz.
- Rapor hazırlanırken referans alınması istenen Proje/Ödev Formatı Classroom'dan paylaşılan formatta olmalıdır.
- Sunumun yapılacağı tarih ve sınıf proje teslim tarihi yaklaştığında duyurulacaktır.
- Öğrenci grupları bağlantıdaki listede yer almaktadır.
https://docs.google.com/spreadsheets/d/1xNZ1D4brgXEeFI_gPp-5-XmQv0ADK6tz/edit#gid=1835081598

Proje 2 Değerlendirme Kriterleri

Proje 2 değerlendirilirken aşağıdaki kriterler göz önünde bulundurulacaktır.

KRİTER	NOT (puan)
1 ve 2. seçenekteki 4 farklı algoritmayı pseudo kod olarak yazmak	20
Algoritmaların karmaşıklık analizleri	20
Algoritmaları gerçekleştirme	10
Karmaşıklık analizlerinin doğrulanması	20
Fonksiyon karmaşıklıklarının raporda karşılaştırılması	10
Rapor	10
Yazılan kodlara hakimiyet	10

ALGORİTMA ANALİZİ PROJESİ

x_1, x_2, \dots, x_n dizisi sıralı olmayan n tam sayıdan oluşmaktadır. $d_{i,j}$ fark değeri, $(1 \leq i < j \leq n)$ olmak şartı ile $|x_i - x_j|$ 'ye karşılık gelmektedir.

- 1) Bu $d_{i,j}$ fark değerlerinden en küçüğünü 2 farklı çalışma zamanında bulan 2 farklı algoritma tasarlayınız.
- 2) Bu $d_{i,j}$ fark değerlerinden en küçük $n/2$ tanesini 2 farklı çalışma zamanında bulan 2 farklı algoritma tasarlayınız.
- 3) 1. ve 2. seçenekler için aşağıdaki görevleri gerçekleştiriniz.
 - Tasarladığınız algoritmaları pseudo kod olarak açıklayınız.
 - Her bir algoritmanın ayrıntılı karmaşıklık analizlerini yapınız. (En iyi ve en kötü durumlar için O , θ , ve Ω karmaşıklığı analizleri.)
 - Tasarladığınız algoritmaları seçeceğiniz bir programlama dilinde n uzunluğundaki diziyi alıp istenen değer/değerleri geri döndüren fonksiyonlar olarak gerçekleyiniz.
 - Fonksiyonlarınızı rastgele üreteceğiniz girdiler ile çok sayıda çalıştırarak karmaşıklık analizinizi doğrulayınız. (Bunun için algoritmanızın temel işlemi fonksiyonların içinde sayılabiliyor olmalı.)
 - Her bir seçenek için fonksiyon karmaşıklıklarını 2 algoritma için karşılaştırınız. Hangisinin daha verimli olduğunu elde ettiğiniz analiz ve doğrulama sonuçlarına dayanarak açıklayınız. Bu değerlendirmelerinizi proje raporunuzun "Test ve Doğrulama" başlığı altında yapabilirsiniz.