

BLM210 PROGRAMLAMA LABORATUVARI I

PROJE 1

PROJE TESLİM TARİHİ: 22.10.2021

BLM110 PROGRAMLAMA

LABORATUVARI I

PROJE 1

PROJE TESLİM TARİHİ: 22.10.2021

Projenin Amacı:

Bu projenin amacı C dilinde dosyalama fonksiyonlarının pratiğinin yapılmasıdır. Bu amaçla ikili dosyaların (binary file) ve metin dosyalarının (text file) kullanıldığı bir uygulama geliştirilecektir.

Bu proje kapsamında yoğun indeks (dense index) yapısı gerçekleştirilecektir. Bu indeks yapısında ikili dosya olarak gerçekleştirilen veri dosyasındaki her kayıt için indeks dosyasında bir girdi saklanmaktadır. İndeks dosyasında kayıtlar <anahtar adres> şeklinde saklanmaktadır. Adres değeri anahtarla ilgili bilginin veri dosyasında hangi offsette saklandığını göstermektedir. Veri dosyasında aynı anahtarla ilgili birden çok kayıt olabileceği için indeks dosyasında aynı anahtara ait birden fazla kayıt bulunabilir. Aşağıdaki görselde bu tür bir indeks yapısı için bir örnek verilmektedir.

Anahtar	Adres
1	0
1	4x
2	x
2	3x
2	6x
3	2x
.	
.	
.	

offset	Anahtar		
0	1		
x	2		
2x	3		
3x	2		
4x	1		
5x	.		
6x	2		
7x	.		
8x	.		

Görselden de anlaşıldığı gibi indeks dosyasındaki kayıtlar anahtara göre sıralı iken veri dosyasındaki kayıtlar sıralı değildir. Böyle bir indeksleme yapısı veri arama işlemlerinin hızını oldukça hızlandırmaktadır. İndeks dosyası kullanılmadan bir kayıt veri dosyasında aranacak olsa tüm kayıtlar dosya baştan sona okunması gerekir. Veri dosyasındaki kayıtların büyük ve çok sayıda kayıt olduğu düşünüldüğünde veri dosyasının baştan sona okunması çok fazla sayıda disk erişimi gerektirir. Oysa ki indeks dosyası kullanıldığında, dosyalar arama anahtarına göre sıralı olduğundan ikili arama yapılabilir. Diğer taraftan indeks dosyasındaki kayıtlar veri dosyasındaki kayıtlara göre çok daha küçük olduğundan indeks dosyasının belleğe aktarılması çok daha az disk erişimi gerektirir.

Bu proje kapsamında veri dosyasında bir okuldaki öğrencilere ait notlar saklanacaktır. Her öğrenci için aşağıdaki struct yapısında bilgi saklanacaktır. Bu yapıda ogrNo, dersKodu ve puan alanları bulunmak zorundadır. Ancak projenizi gerçekleştirirken ihtiyaç duyabileceğiniz başka alanlar da tanımlayabilirsiniz. Bir öğrenci için birden çok kayıt olabileceğini göz önünde bulundurunuz. Veri dosyasının bir ikili dosya (binary file) şeklinde gerçekleştirilmelidir.

```
struct kayıt{
    int ogrNo;
    int dersKodu;
    int puan;
    .
    .
}
```

Gerçekleştirilecek fonksiyonlar:

- indexDosyasiOlustur: Bir veri dosyasının henüz indekslenmediğini kabul edip veri dosyasındaki kayıtlar için bir indeks dosyası oluşturulacaktır.
- kayıtEkle: Veri dosyasına yeni bir kayıt eklenecektir. Yeni kayıt her zaman veri dosyasının sonuna eklenmelidir. Kayıt eklendikten sonra indeks dosyası güncellenmelidir.
- kayıtBul: Verilen bir anahtar için veri dosyasındaki ilk kayıt gösterilecektir. İlgili anahtar için birden çok kayıt bulunabileceği için aranan kayıt bulunana kadar anahtara ait kayıtlar listelenecektir. İndeks dosyasında anahtara ait ilk kaydın bulunmasında ikili arama (binary search) algoritması kullanılacaktır.
- kayıtSil: Verilen bir anahtar için ilgili doğru kayıt bulunacak (kayıtBul işlemindeki gibi) ve bu kayıt silinecektir. Silme işlemi indeks dosyasına da yansıtılacaktır.
- kayıtGuncelle: Bir anahtar için aranan kayıt bulunup sadece puan alanı güncellenecektir.
- veriDosyasiniGoster: Veri dosyasındaki tüm kayıtları listeler.
- indeksDosyasiniGoster: İndeks dosyasındaki tüm kayıtları gösterir.
- indeksDosyasiniSil: İndeks dosyasını diskten siler.

Önemli notlar:

- Proje C dili kullanılarak gerçekleştirilecektir.
- Veri dosyası binary file olmalıdır.
- İndeks dosyası text file olmalıdır.

- İndeks dosyasının ve veri dosyasının boyutları hakkında herhangi bir kısıt yoktur. Eğer bu dosyalar dizilere aktarılacaksa dizilerin boyutları dinamik olarak belirlenmelidir.
- Ödevinizi teslim ederken tanımladığınız struct kullanılarak oluşturulmuş 50 kayıtlık veri dosyası da teslim ediniz.

Sunum: Proje sunumu E-Destek üzerine yükleyeceğiniz projenizdeki kodlar indirilerek alınacaktır. Bu nedenle E- Destek üzerine yükleyeceğiniz projenin doğruluğundan emin olunuz.

Açıklama:

Ödev Teslimi

- Rapor ieee formatında (önceki yıllarda verilen formatta) 4 sayfa, akış diyagramı veya yalancı kod içeren, özet, giriş, yöntem, deneysel sonuçlar, sonuç ve kaynakça bölümünden oluşmalıdır.
- Dersin takibi projenin teslimi dahil edestek.kocaeli.edu.tr sistemi üzerinden yapılacaktır. edestek.kocaeli.edu.tr sitesinde belirtilen tarihten sonra getirilen projeler kabul edilmeyecektir.
- Proje ile ilgili sorular edestek.kocaeli.edu.tr sitesindeki **forum üzerinden** Arş.Gör. Muhammed Ahmet Demirtaş veya Arş.Gör. Ayşe Gül EKER'e sorulabilir.
- Sunum tarihleri daha sonra duyurulacaktır.
- Sunum sırasında algoritma, geliştirdiğiniz kodun çeşitli kısımlarının ne amaçla yazıldığı ve geliştirme ortamı hakkında sorular sorulabilir.
- Kullandığınız herhangi bir satır kodu açıklamanız istenebilir.