KOCAELİ ÜNİVERSİTESİ

BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

PROGRAMLAMA LABORATUVARI I

ÖĞRENCİ KAYIT

FAİK PİRİNÇCİ-ALPER TALHA KÜÇÜK

[190202108@kocaeli.edu.tr](mailto:190202108@kocaeli.edu.tr) / [180202034@kocaeli.edu.tr](mailto:180202034@kocaeli.edu.tr)

ÖZET

Proje dosyasını ilk gördüğümüzde aklımızda birkaç tane soru işareti vardı. Bunların ilki binaryfile’a dosya yazma ve okuma işlemlerinin nasıl yapılacağıbinarysearch’e nasıl bir ekleme yapmalıyız ki aynı kayıtı bulana veya bulamayana kadar aynı anahtara sahip kayıtları göstersin , kayıt ekleme kısmında binaryfile’ın sonuna kayıt ekleme işlemi yapılması isteniyordu. Standart bir txt file gibi “a” modunda açıp ekleyebilir miyiz , eğer dizi oluşacaksada dinamik olması isteniyordu. Bunun için kayıt sayılarını nasıl çekeceğizdi nasılgerçeklemeliyizbu konuyu tam olarak düşünemedikbaşta , araştırmalarımızı yaptıktan sonragidilebilecek bir yol bulmaya çalıştık Zamanımızı verip uğraştıktan son yapılamayacak bir proje değildi. Fakatbaşlangıçta büyük bir sorun gibi gözüktü çünkü birazını kafamızda tahmin edip gidiş yolumuzu hesap edebiliyorduk, fakat kesinliğinden emin değildik buda bizi daha da düşündürüyordu. Bunun ardından çevremizdeki bazı kod bilen insanlardan yardım istedik ama direkt olarak bizim ihtiyacımız olan bilgiyi veremiyorlardı en sonunda koda giriş yaptık bir kaç hatanın ardından binary file hakkında dosyalar bulup başkaları nasıl yapmışlar inceledik , binarysearchtree ,küçükten büyüğe sıralama algoritmaları , distinct algoritmalarını inceledik sonunda da anlayıp koda döktük projemizin özgün değeri şuanda büyük uğraşlar sonucunda ortaya çıkan bu proje çabalarımız ve çalışmalarım açısından çok zaman almıştır.

YALANCI KOD

Kullanıcıdan değerini secimeal ;

Seçim 1’e eşitse indexdosyasiolustur() git ;

Seçim 2’ye eşitse ;

Ogr tanımla ve kullanıcıdan değer al ;

Derskodu tanımla ve değeri al ;

Puan tanımla ve kullanıcıdan değer al ;

Bir structa bu değerleri ata ;

Kayıtekle fonksiyonuna bu structıgönder ;

Seçim 3’e eşitse ;

ogr no için değer al ;

ders kodu için değer al ;

puan için değer al ;

Bu bilgileri kayıtbul fonksiyonuna gönder ;

Seçim 4’e eşit olursa ;

Kayıtsilfonksionunagit ;

Seçim 5’e eşit olursa ;

Kayıtgüncelle fonksiyonuna git ;

Seçim 6’ya eşit olursa;

Veridosyasinigoster() fonksiyonuna ‘ git;

Seçim 7’ye denk olursa ;

İndexdosyasinigoster() fonksiyonuna git ;

Seçim 8’ye denk olursa ;

İndexdosyasinisil() fonsiyonuna git ;

Seçim 9’a eşit olursa ;

Veridosyasinisayisal() fonksiyonuna git ;

Seçim 10’a eşit olursa ;

İndexdosyasinisayisal() fonksiyonuna git ;

GİRİŞ

Projemizde Öncelikle bizden istenen 2 farklı dosyalama çeşidi olan standart kullandığımız txt file ve binary file çeşitlerini kullanmamız. Tabi bunlara kullanmadan önce öğrencilerin numarası ders kodu , puanıda bu dosyaların içerisinde olması isteniyor binaryfile’da sıra farketmeksizin anahtar ogr no olarak alınacak diğer tarafların adresleri tutulacaktır.bu kısımlar ise struct yapısı içerisinde bulunmalıdır.Biz başlangıçta veridosyası oluşturma bölümünü fonksiyonda denediysekte adreslerin gösterdiği yerde doğru değerleri bulamadık bu yüzden mainde tanımladık .Fonksiyon kısmında bizden istenilen bazı şeyler var bunlara değinmek gerekirse öncelikle kayitbul kısmında ilk kayıt binarysearch algoritmasıyla bulunması zorunlu ve o kayıt bulunana kadar aynı anahtara sahip diğer kayıtların gösterilmesi istenmekte , kayıtekle kısmında bizden istenen temel şey veri dosyasına eklenecek kaydın veridosyasının sonuna eklenmesi , yada kayitsil ve kayıt güncelle fonksiyonlarının ardından indexdosyasınında güncellenmesi istenmektedir. Bu ve bunlar gibi tüm isterlerin ardından index dosyasının silinmesi bizden isteniyor.Aynı zamanda Yapmış olduğumuz bu projenin bize göre 2 amacı vardır.İlk olarak kod yazımında bizi geliştirerek ilerde meslek sahibi olmamız yolundaki gereken adımları atmamızı sağlamak ve aynı zamanda bu sene gördüğümüz geçen sene gördüğümüz C programlama dersi üzerine öğrenmiş olduğumuz bilgileri unutmamızı engelleyip hafızamızı dinç tutmak tabiki bunun yanında farklı amaçlarıda var bizim öğrendiğimiz dosya okuma yazma , struct gibi bilgimizi kullanıp bu bilgilerimizi bir projede bir problemde çözüm olarak nasıl kullanmamız gerektiğini bize en iyi öğretme şekli olan proje şekliyle öğretiliyor.

diğer amacı aynı zamanda farklı problemlerle bizi karşılaştırıp farklı düşünme , sorunlarla 1 e 1 temas kurmamızı sağlayıp kendimizi geliştirmek gibi daha artırılabilir amaçları bulunmaktadır.

Erişilen çıktılar analizlerin tam olarak söylenmesi gerekirse projemiz istenilenherşeyide başarı ile gerçekleştirmektedir.Lakin biz projemizin çalıştığından ne kadar emin olsaktabazen realtestlerde anlık sıkıntılar çıkabilmekte bu durumlardan dolayı kuşkulu bir cevapta vermiş olduğumuz bir gerçektir.

TEMEL BİLGİLER

Projemizi yaparkenyani kodumuzu yazarken kullandığımız sadece dev c,internettir.

DİĞER BÖLÜMLER

Bölüm 1

Projemizin başlangıcında veridosyasını bir fonksiyonda oluşturduk fonksiyonda olduğunda ve veri dosyasının bir kısmı adres olarak tutulduğundan adreslerin gösterdiği yerlerdeki değerlere erişme konusunda sıkıntı çektik açıkçası bunun fonksiyonda oluştuğunda belleğe tam olarak yazılmadığını düşünüyoruz. Bunun çözümü olaraktaveridosyasını mainde oluşturduğumuzda sıkıntılar giderilmiş oldu. İlk önce main fonksiyonumuzda veri dosyasını oluşturduk ardından kullanıcı sürekli istediği şeyleri yapabilsin diye bir while(1) döngüsü içerisinde kullanıcıdan değer alarak oluşturmamızı istedikleri fonkyonları burada o değere göre çağırdık ardından bizden istenilen fonksiyonlara ekleme yapma ihtiyacı gördüğümüzden dolayı datacount isimli bir fonksiyon oluşturduk bizden oluşturulan dizilerin dinamik tanımlanması istenmekteydi. Bu datacount isimli fonksiyon sayesinde veridosyasında toplam kaç kayıt ver bize bu değeri döndermesini sağladık fonksiyonun içeriğine değinmek gerekirse veri dosyası freadle okunma işlemi sırasında bir struct dizisinin içerisine atama yapıyoruz. Fakat şöyle bir sıkıntı mevcut ki veri dosyasının sonuna geldiğinde ve tekrar veri çekmek istediğimizde döngü sayısına bağlı olarak sürekli en son kayıtı bize veriyor bizde sürekli dönecek bir döngü içerisinde bir önceki kayıt bir sonrali kayıta eşit mi diye kontrol ediyoruz bu sayede veridosyasındaki toplam kayıt sayısına erişiyoruz. Ardından indexdosyası oluşturma fonksiyonuna geliyor burada yapılan dinamik olması istendiğinde veri dosyasındaki kayıt sayısını bulup o kadar içerisinde struct alabilecek bir struct dizisi oluşturmak ardından o kadar dönecek bir döngüyle tüm kayıtları struct dizisinin içerisine alıyoruz . Sonrasında normal internetten gördüğümüz bir sıralama algoritmasını uyguluyoruz . Bu algoritma sayesinde struct dizisinin içerisindeki kayıt ogrno ya göre küçükten büyüğe sıralanıyor ardından artık sıralanmış kayıtları indexdosyasının içerisine basıyoruz. Kayıt ekleme fonksiyonunda öncelikle yine main kısmında bilgiler alınıp ona göre bir struct oluşturulup kayıtekle fonksiyonuna bu struct geliyor gelen structı veri dosyasını “ab” modunda açtıktan sonra en sonuna ekliyoruz ve ardından index dosyasını güncellemek için tekrar veri dosyası kayıt sayısını alıp o kadar struct dizisi oluşturuyoruz. Ardından döngü kayıt sayısı kadar dönüyor ve veri dosyasındaki tüm kayıtları struct dizisine çekiyoruz. Sonrasında bu diziyi index dosyası oluştur kısmındaki gibi küçükten büyüğe sıralayıp indexdosyasının içerisine basıyoruz. Sonrasında Kayıtbulfonksyonunu gerçekleştirdik bu fonksiyonda öncelikle main kısmında kullanıcının bulmak istediği kayıt değerleri alınıyor ve kayıtbul fonksiyonuna gönderiliyor ardından datacount’la kayıt sayısını buluyoruz. O kadar struct oluşturuyoruz ve index dosyasının içerisindeki sıralı verileri bu struct dizisinin içerisine atıyoruz.Sonrasındabinarysearch uygulanıyor eğer kayıt bulunursa breakle döngü kırılıp flag 1’e eşitleniyor bulunamadığı her index’teogr no aynı ve diğer kısım farklı mı kontrolü yapılıyor eğer öyleyse o anki index bir diziye atılıyor ve o diziye 10 atama yapıldığında yine break ile döngüyü kırıyoruz burdaki temel nokta şudur ki bahsemiş olduğumuz kısımda döngünün içerisine sürekli aynı değerleri atıyor aynı indexleri sonrasında bu aynı değerlerin bulunduğu diziyi internetten araştırarak gördüğümüz bir distinct algoritmasına sokuyoruz ve sadece aynı anahtara sahip fakat istenilen olmayan kayıtları görüntüleyebiliyoruz. Kayıt silme fonksiyonun çalışma mantığı tam olarak şöyle istenilen kayıt bilgileri alınıyor ardından o kayıt var mı varsa hangi indexte onu buluyoruz ve sonrasında tekrardan yazma işlemi yaparken o indexe geldiğimizde yazma işlemi yapmıyoruz diğer indextendevamediyor bu sayede silme işlemini gerçekleştirmiş oluyoruz. Yine ardından kayıtekle kısmında bahsettiğim gibi index dosyası güncelleniyor.Kayıt güncelle kısmında kullanıcıdan ogr no ders kodu ve yeni puan alınıyor ardından tekrar veri dosyasının içerisindeki kayıtlar bir struct dizisine atanıyor bu kayıtlarda verilen bilgilere eşleşme varsa puan kısmına yeni puanı atıyoruz. Ardından tekrar yazdırıyoruz .ve bir önceki kısımlardaki gibi index dosyası güncelleme işlemleri yapılıyor sonrasında verigöster kısmı için veri dosyası kayıtları bir structa çekilip bu kayıtlar bir for döngüsü ile basılıyor index dosyası içinde aynı şekilde kayıtlar structa çekilir for döngüsü ile basılıyor index dosyası sil fonksiyonu için remove kullanıyoruz eğer remove’dan dönen değer 0 ise dosya silindi kısmına gidiyor.

SONUÇLAR

Projemiz çalışmalar ve çabalar ile yapılmıştır.Ve GözlemlenenHer hangi bir hata bulunmamaktadır.

Proje başlangıcında eksik olan yanlarımız yok denecek kadar azdı binary file , veridosyasınınsonuna eleman ekleme gibi şeyler zor gibi gelmişti.onu nasıl yapacağımız kısmını düşünmekte sıkıntı çektikbunun sonucunda direkt olarak projemizin kodlarını yazmaya başlayamadık araştırmamızın başlangıcında yakınımızdaki kod bilen insanlara iletişime geçtiysekte bir katkısı pek olmadı internetten araştırmalar yapıp bunları öğrendik ve koda geçirdik. Ve de herhangi bir sıkıntı sorun bulunmamaktadır.

KAYNAKÇA

https://bilgisayarkavramlari.com/2009/12/21/ikili-arama-algoritmasi-binary-search-algorithm/

https://web.cs.hacettepe.edu.tr/~maydos/Docs/c/dosyalar.pdf

<https://www.faceprep.in/c/distinct-elements-in-an-array/>

https://www.algoritmaornekleri.com/c-plus-plus/c-dizi-elemanlarini-kucukten-buyuge-siralama/