

BLG 221

Veri Yapıları

2010 Güz

Ödev 1

Son Teslim Tarihi : 07.10. 2010

Ad Soyad : Ozan Arkan Can

Numara : 040090573

Sınıf Öğitmeni : Gülşen ERYİĞİT

Asistan: Ahmet Aycan ATAK

CRN: 10950

1-Giriş

Bu ödevin amacı: derste gösterilen telefon uygulamasında dosya::arama fonksiyonu arayıp bulduğu kriterlere uygun kayıtları sadece ekrana bastırıyor ama bunların hangi kayıtlar olduğunu bir veri yapısında tutup kendini çağıran fonksiyona döndüremiyor.

Bu nedenle kayıt_arama fonksiyonu da bulduğu kayıtları ekrana yazıp kullanıcıdan gördüğü numarayı yazmasını istiyor. Silme ve güncelleme işlemlerini de kullanıcının girdiği bu numaranın doğru olduğuna güvenerek yapıyor. Ancak kullanıcının ekranda yazan değerlerden farklı bir kayıt girip girmediği de kontrol edilemiyor.

Kriterlere uyan tek bir kayıt olsa bile kullanıcıdan ekranda okuduğu numarayı girmesi isteniyor.

Bu problemi düzelten ve kullanıcının silme ve güncelleme işlemlerinde ekranda olmayan bir sayı girdiğini denetleyen şekilde programı değiştirmektedir.

2-Problem için geliştirilen çözüm

Kayıt sıralarının bir veri yapısında tutulmuyor oluşuna dosya.arama fonksiyonun çalışması sırasında aranılan kritere uygun kayıtlar bulunduğu sırada bu kayıtların sıraları (sayac) kayıtno.txt dosyasına yazılarak bir çözüm getirilmiştir.

dosya.sil ve dosya.guncelle fonksiyonlarının çalışması sırasında aranılan kriterlere uyan tek bir kayıt olduğunda kullanıcıdan yine de kayıt numarası isteniyordu. Bunun yerine kullanıcıya bulunan kayıt gösterilip silmek (güncellemek) isteyip istemediği soruluyor. Evet cevabı alınırsa kayıtlar.txt dosyasından 1 tane olan kayıt numarası alınıp dosya.sil (dosya.guncelleme) fonksiyonlarına parametre olarak döndürülüyor. Hayır cevabı alınırsa hiç bir şey döndürülmeden menüye geri dönülüyor.

dosya.sil ve dosya.guncelle fonksiyonlarının çalışması sırasında aranılan kriterlere uyan birden çok kayıt olduğunda kullanıcıdan kayıt sırasının girilmesi isteniyor fakat doğru bir giriş yapıp yapılmadığı kontrol edilmiyor. Bunu önlemek için bu esnada kayıtno.txt dosyasında bulunan kayıt sıraları okunup bir diziye aktarılıyor. Kullanıcın girdiği kayıt sırası bir döngü içinde bu dizi elemanlarıyla karşılaştırılıyor. Eğer bir eşitlik söz konusu olduğunda dosya.sil (dosya.guncelle) fonksiyonuna kullanıcın girdiği değer parametre olarak gönderiliyor. Break komutuyla döngüden çıkılıyor.

Eğer tüm döngü boyunca bir eşitlik söz konusu olmazsa döngü bittikten sonra kullanıcıya girdiği kayıt sırasının eşleşmediğine dair uyarı mesajı veriliyor.

Not: Dizin kullanılması max eleman sayısından dolayı karşılaştırma yapılacak kayıt sayısında bir sınırlamaya sebep olmuştur.

3- Veri Yapıları ve Değişkenler

"kayitlarno.txt" //Kayıt sıralarının tutulduğu dosya veri yapısı

FILE *yenidosya; // Kayıt sıralarını tutan dosya için pointer

int kayitno_d[MAX];

/* Silme işlemi için onay alınırken girilen kayıt sırasının arama neticesinde bulunan kayıtlar içerisinde olup olmadığını kontrol etmek için oluşturulan dizi*/

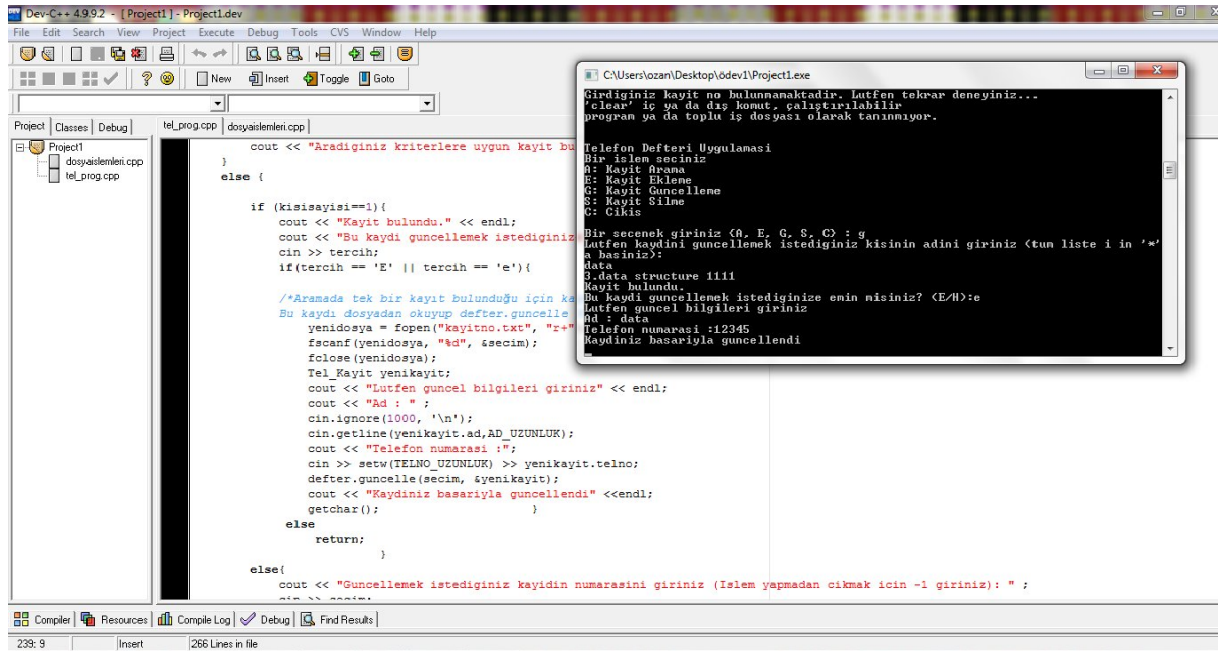
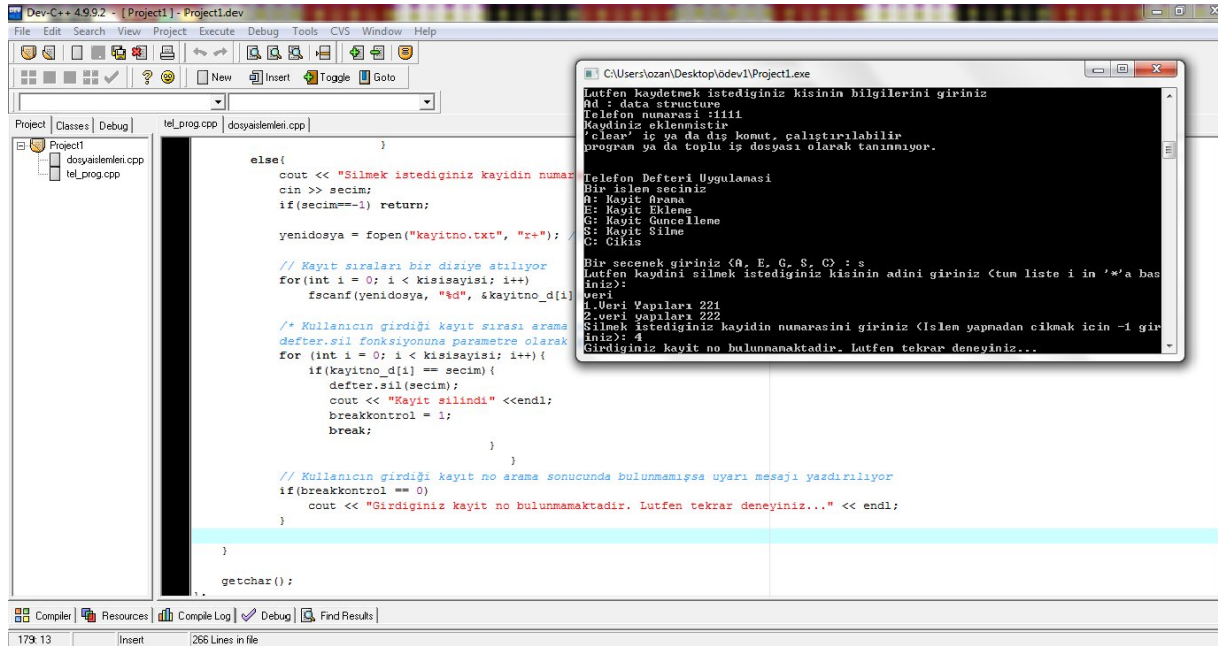
char tercih; // Kullanıcıdan evet/hayır girişi almak için değişken

int breakkontrol = 0; // Kullanıcının girdiği kayıdın eşleşmemiş olması durumuyla ilgili bir değişken

4-Geliştirme ve Çalıştırma Ortamları

MS Windows

Kaynak kodu düzeltmede, derleme aşamasında ve çalıştırmada Dev-C++ kullanılmıştır.



5-Sonuç

Birden çok kaynak dosyası ve header dosyası kullanılan projelerin anlaşılması için iyi bir deneyim aynı zamanda veri yapıları dersi için gerekli bir uygulama oldu.