**1-ÖNSÖZ**

*Proje bir otobüs firmasının bilet satış uygulamasıdır. Program Sefer ve Satış işlemleri olmak üzere iki bölümden oluşmaktadır.*

*Sefer işlemlerinde sefer no , güzergah , tarih , saat ,yolcu kapasitesi , bilet fiyatı ,kaptan, bilgileri kullanıcı tarafından girilir. Sefer ekleme, sefer silme, kaptan değiştirme , sefer gelir hesaplama ve sefer listeleme işleri yapılır.*

*Satış işlemlerinde bilet satışı , bilet rezervasyon ,bilet iptal işlemleri yapılır.*

**2-GİRİŞ**

*Yazılım otobüs firmasına bilet satış işlemleri yapan uygulamadır. Yazılım dinamik yapıda olup sınırsız sefer ve otobüs bilgisi tutabilmektedir.*

**3-PROBLEM ÇÖZÜM VE UYGULAMA**

**3.1 Probleme Çözüm Yaklaşımı**

*Yazılımın sınırsız sefer ve otobüs bilgisi tutabilmesi için dinamik yapı kullanıldı. Bu yapı çift yönlü bağlı liste kullanılarak sağlatıldı.*

**3.2 Yazılım Mimarisi**

***Fonksiyonlar***

***main()***

*main fonksiyonu çalıştırıldığında ilk olarak sistem tarihini global olarak tanımlanmış olan bugun adlı katara kopyalıyor, daha sonra basla adlı fonksiyon çağrılar data dosyasındaki verilere göre sefer listesi okunuyor son kullanılacak olan sefer numarası anlıyor. Secenek olarak tanımlanmış olan karakter değişkeni ise kullanıcının menüler arası gezerken yaptığı seçimleri saklıyor.*

*İlk olarak kullanıcıya ana menü seçeneklerini vererek kullanıcının secim yapmasını istiyor. Girilen secime uygun fonksiyon çağrılarak işlem tamamlanıyor.işlem sonrası kullanıcı alt menüye inebiliyor veya programı sonlandırabiliyor.*

***Sefer\_ekle();***

*Fonksiyonun log kaydını tutmak için FILE \*log olarak tanımlanmış olan dosya işaetcisi a+ modunda açılıyor. Eklenecek sefer seferler struct ının türünde yer adlı değişken tanımlanıyor daha sonra tanımlanan bu değişken için malloc fonksiyonuyla yer istenerek istenen bu yere sefer bilgileri giriliyor ve sefere ait koltuk bilgileri bu fonksiyon ile otomatik olarak atanıyor. Daha sonra yer alanınadaki bu bilgiler sıralı listenin sonuna eklenerek log kaydı yapıldıktan sonra eklenen sefer dosyaya yazılıp güncellenerek fonksiyon sonlandırılıyor.*

***Sefer\_sil();***

*Fonksiyon log kaydı tutmak için FILE \* log dosya değişkeni a+ modunda açıyor*

*Daha sonra kullanıcıdan silinecek sefer numarasını isteyerek girilen sefer numarasını numara değişkenine atıyor sil değişkeni ile seferler sıralı listesinin ilk elemanına erişiyor ve while döngüne sokulan sil değişkeni sıralı listenin sonuna gelene kadar değişken bir sonraki elemana atanarak o andaki sefer numarası kullanıcıdan alınıp numara değişkenine atılan numarayla karşılaştırılıyor eğer iki numara eşitse koltuk durumunu denetliyor. Eğer kltuk durumu dolu yani 2 ise bilet satıldığı için sefer silinemez çıktısını veriyor. Durum 2 değilse yani boşsa sefer sıralı listeden siliniyor.sıralı listeden silinirken silinecek seferin adresi alınıyor eğer bu adres sıralı listenin ilk adresi ise sıralı listenin ilk adresi silinen seferden sonraki seferin adresini tutuyor. Eğer ilk değilse sefer listeden ayrılıyor. Log kaydına yapılan işlemi yazdıktan sonra silinen seferin dosyanı güncelleyen dosya\_guncelle() fonksiyonu çağırılıyor log kaydını tutan dosya fclose() fonksiyonuyla kapatılarak fonksiyon sonlandırılıyor.*

***sefer\_listele()***

*Bu fonksiyon henüz tarihi geçmemiş olan seferleri listelememizi sağlıyor. Seferler structının türünde deklare edilen listele işaretcisini sefer listesinin ilk elemanına eşitliyor. Bu işaretçi daha sonra while döngüsüne sokularak iste sonuna gelene kadar tüm sefer bilgilerini ekrana yazıyor. Log kaydı yapıldıktan sonra fonksiyon sonlandırılıyor.*

***sefer\_sayisi()***

*Bu fonksiyon tarihi geçmemiş toplam sefer sayısını hesaplıyoruz. Fonksiyon çağrıldığı zaman log kaydını tutmak için log dosya işaretcisinde log dosyası açılıyor seferler \*say değişkenine sıralı listenin ilk elemanı atanıyor daho sonra bu değişken while() döngüsüne sokularak liste sonuna gelene kadar her sefer için öceden tanımlanan sayı değişkenini 1 artırıyor ve toplam sefer sayısı yazdırılıp log kaydı yapılarak fonksiyon sonlandırılıyor*

***sefer\_no\_guncelle()***

*fonksiyon data dosyasında tutulan sefer numarasını güncellemek için kullanılıyor data dosyası r+modunda açıldıktan sonra dosyadaki sefer nosu yerine o an geçerli olan sefer nosu yazdırılıp program güncelleniyor.*

***sefer\_menu()***

*sefer işlemlerini hangi seçenek ile belirleneceğini ekrana listeleyen fonksiyon.*

***koltuk\_listele(struct koltuk\_listeleri \*kol\_listele)***

*bu fonksiyon kendisine gönderilen koltuk listesinin ilk elemanını while döngüsüne sokuyor ve koyluk listesini son elemanına gelene kadar her koltuğun bilgileri listelenip döngü sonlandırılarak fonksiyon tamamlanıyor.*

***karsilastir(char tarih1[],char tarih2[])***

*karsilastir fonksiyonu kendisine gelen iki tarihigün ay ve yıl olarak parçaladıktan sonra bunları tek tek atoi fonksiyonu ile sayıya çeviriyor ve tarihler struct ına yerleştiriyor. Daha sonra struct taki bilgiler if yapılarıyla karşılaştırıldıktan sonra eğer 1. Tarih 2. Den büyükse 0 2. Tarih 1. Den büyükse 2 tarihler eşitse 0 gönderiri.*

***kaptan\_degistir()***

*kullanıcıdan kaptanını değiştirmek istenilen seferin numarası alınarak bu numaralı sefer listeden bulunuyor sefer bulunduktan sonra yeni kaptanın adı kullanıcıdan isteniliyor ve eski katan adıyla değiştiriliyor.*

***gelir\_hesapla()***

*bir sefer ait koltuk durumları göz önüne alınarak seferden elde edilen toplam gelirin hesaplanmasını sağlıyor sefer numarası kullanıcıdan alındktan sonra bu seferin koltuk listesinde her koltuk durumu değerlendiriliyor eğer koltuk satışı yapılmışsa bu koltuğun bilet fiyatı toplam gelire ekleniyor log kaydı yapılıp fonksiyon sonlandırılıyor.*

***gecmis\_oku()***

*Bu fomksiyon ile data dosyasında bulunan tarihler alınarak karsilastir fonksiyonu ile*

*Günün tarihi ile karşılaştırılıyor karşılaştırma sonucunda dosyadan okunan tarih günün tarihinden geri ise bu tarihe ait dosya diskten okunarak sıralı listenin başından başlanarak listeye ekleniyor önceki listenin kaybolmaması için bu listenin baş son elemanları gecici\_ilk ve gecici\_son değişkenlerine atanıyor geçmiş seferler listelendikten sonra bu değişkenler ilk ve son işaretcilerine tekrar ekleniyor.*

***dosya\_yaz(struct seferler \*yaz)***

*Bu fonksiyon kendisine gelen seferi mensubu olduğu dosyaya yazıyor eğer böyle bir dosya yoksa yani o tarihteki ilk sefer kendisi ise bu dosya oluşturuluyor. Dosya a metodu ile açıldıktan sonra bu sefere ait bilgiler ve bu seferin koltuk listesine dosyaya eklendikten sonra fonksiyon sonlanıyor.*

***dosya\_oku(char isim[])***

*Bu fonksiyon kendisine gelen tarih adındaki dosyayı okuyor. Gelen tarihe .txt uzantısını ekledikten sonra ad katarına atıyor. Dosya sonuna gelene kadar devam eden bir while döngüsüne sokuluyor ve dosyadaki her sefer sırasıyla okunuyor sefer bilgileri okunduktan sonra bu sefer ait koltuk bilgileri okunuyor ve fonksiyonun başında oluşturulan sıralı listeye atanıyor en sonunda bu listenin ilk elemanı serfler listesinde uygun yere konularak fonksiyon sonlandırılıyor.*

***dosya\_guncelle(char tarih[])***

*Bu fonksiyon kendisine gelen tarihin sonuna txt uzantısını ekledikten sonra elde edilen dosya adlı dosyanın güncellenmesini sağlıyor oluşturulan addaki dosya w metoduyla açılıyor açıldıktan sonra seferler listesindeki her seferin tarihi açılan dosya tarihi ile karşılaştırılarak aynı tarihli ise dosyaya ekleme işlemi yapılıyor listenin sonuna gelindikten sonra yazma işlemi sonlandırılıp fonksiyon tamamlanıyor.*

***bilet\_islem(int secenek)***

*Bilet işlemlerini yapmaya yarayan bu fonksiyon seçenek argümanına gelen değer 0 ise rezervasyon 1 ise satış 2 ise iptal işlemini gerçekleştiriyor. Kullanıcıdan işlem yapılmak istene sefer numarası alınıyor while döngüsü ile sefer bulunduktan sonra işlem yapılmak istenen koltuk numarasının seçilmesi isteniyor bulunan koltuk numarasına seçilen işlem yapıldıktan sonra dosya güncellenip log kaydı yapıldıktan sonra fonksiyon sonlanıyor.*

**ana\_menu()**

*Bu fonksiyon ana menüyü düzenleme işlemi yapar. Windows konsol fonksiyonu kullanarak konsol renklendirilir. Yazdırma ve boşluk fonksiyonlarıyla menü düzenlenir.*

**bilet\_menu()**

*Bu fonksiyon bilet menüsünü düzenleme işlemi yapar. Windows konsol fonksiyonu kullanarak konsol renklendirilir. Yazdırma ve boşluk fonksiyonlarıyla menü düzenlenir.*

**bosluk()**

*Düzenlemelerin daha okunaklı olması için 20 karakterlik boşluk koyar.*

**bugun\_tarih()**

*Sistem tarihini belirleyip günün tarihini kullanmak için kullanılır.*

**basla()**

*Bu fonksiyon basta sefer no ve sefer tarihlerinin bulunduğu data.txt dosyasını kontrol eder yoksa oluşturur.Olusturulan data.txt dosyasına sefer no ve tarihleri yazdırır.Data.txt den okunan tarih ile bugun\_tarih fonksiyonu ile gelen bugunun tarihi karşılaştır fonksiyonu ile karşılaştırılır.data.txt ten okunan tarih bugünün tarihi veya bugünün tarihinde sonraki tarih ise tarihi bağlı listeye atar daha sefer\_no\_güncelle fonksiyonu ile dosya güncellenir.*

**3.3 Yazılım Çalıştırılması**

*1) 130201001-130201047.cpp dosyasına çift tıklanır*

*2) F11 tuşu ile 130201001-130201047.cpp dosyası derlenir ve çalışır.*

*3) Uygulamanın ana menüsü açılır.*

*4)Ana Menüde*

*a) (k ve K) programı kapatır.*

*b) (1) Bilet İşlemlerini açar.*

*c) (2) Satış İşlemlerini açar.*

*5) Bilet İşlemlerinde*

*a) (A veya a) Ana Menüye döner.*

*b) (1) Sefer Ekle İşlemi yapar. Burada sefer bilgileri girilir ve sefer eklenir.*

*c) (2) Sefer Sil İşlemi yapar. Burada listelenen seferlerden seçilen sefer numarası ile sefer silinir.*

*d) (3) Sefer Listeleme işlemi yapar. Eklenen seferler listelenir.*

*e) (4) Geçmiş Seferleri Listeleme işlemini yapar. Sistem tarihinden önceki tarihlerdeki seferleri listeler.*

*f) (5) Sefer Geliri Hesaplama işlemi yapar. Listelenen seferler arasından seçilen seferin sefer numarası girilerek gelir hesaplanır.*

*g) (6) Tüm Sefer Sayısı işlemi gerçekleştirir. Eklenen seferlerin sistem tarihi ve o tarihten sonraki seferlerin sayısını verir.*

*h) (7) Kaptan Değiştir işlemini gerçekleştirir. Listelenen seferler arasından seçilen seferin sefer numarası girilerek o seferini kaptanını değiştirme işlemi yapar.*

*6) Satış İşlemleri*

*a) (A veya a) Ana Menüye döner****.***

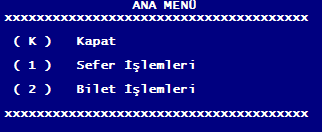
*b) (1) Bilet Satiş işlemi yapar. Listelenen seferler arasından seçilen seferin sefer numarası girilir. Daha sonra kişisel bilgiler ve koltuk numarası girilerek bilet alma işlemi gerçekleştirilir.*

*c) (2) Bilet Rezervasyon işlemi yapar. Listelenen seferler arasından seçilen seferin sefer numarası girilir. Daha sonra kişisel bilgiler ve koltuk numarası girilerek rezervasyon işlemi yapılır.*

*d) (3) Bilet İptal işlemi yapar. Listelenen seferler arasından seçilen seferin sefer no girilir ve bütün koltuklar listelenir. İptal edilecek koltuk numarası girilir ve iptal işlemi yapılır.*

**3.4 Yazılım Sonuçları**

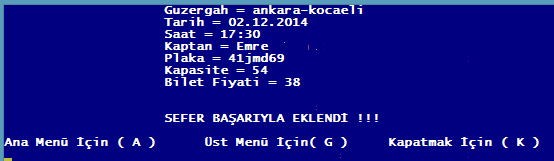
*1.Ana Menü*

**

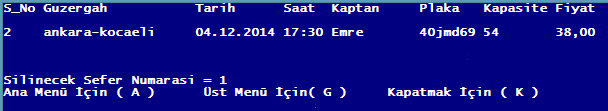
*2.Sefer İşlemleri*

**

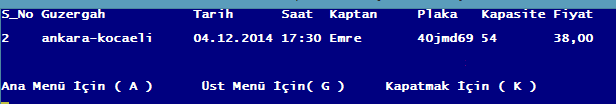
*2.1.Sefer Ekle*

**

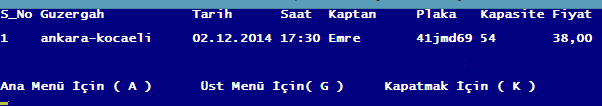
*2.2..Sefer Sil*

**

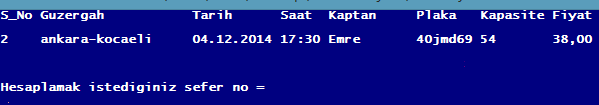
*2.3.Sefer Listeleme*

**

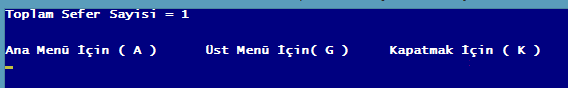
*2.4.Gecmiş Seferleri Listele*

**

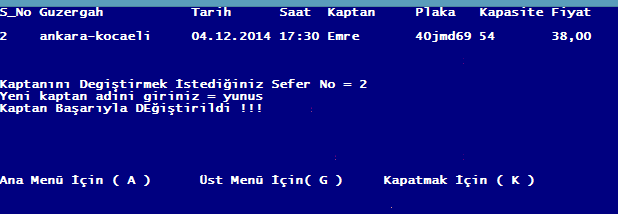
*2.5.Sefer Geliri Hesapla*

**

*2.6.Tüm Sefer Sayisi*

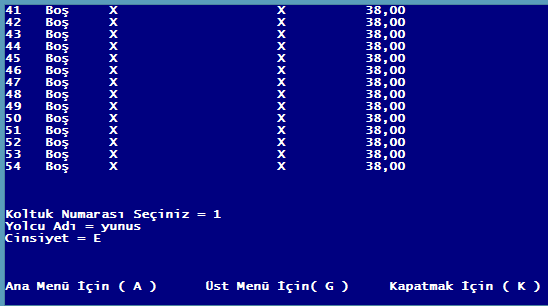
**

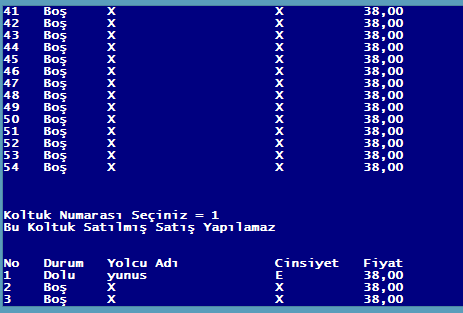
*2.7.Kaptan Degiştir*

**

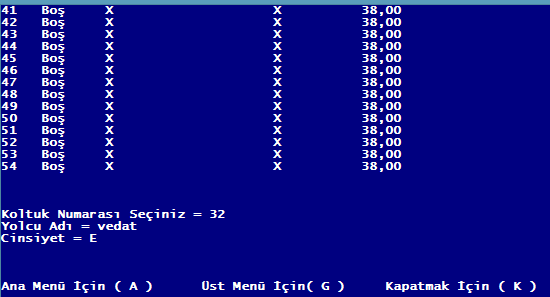
*3.Satiş İşlemleri*

*3.1Bilet Satis*

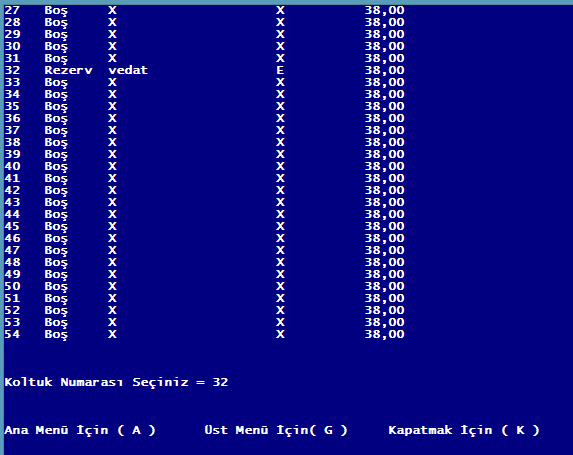
**

**

*3.2 Rezervasyon*

**

*3.3 Bilet İptali*

**

**4-KAYNAKLAR**

1)Bogaziçi University Library Cataloging in Publication Data

C && Java ile Veri Yapılarına Giris/ Olcay Taner Yıldız. 299p. ;23 cm

2)İşte C Programlama Dili /Rifat Cölkesen. –İstanbul:Papatya Yayıncılık,2013

Xiv,376s. ; 24 cm