Bağlı Liste Uygulaması

180201097 Meleknaz ABLAK

170201033 Rabia GÜNEŞ

Bilgisayar Mühendisliği Bölümü

Kocaeli Üniversitesi

mhnds\_rabiagunes@hotmail.com , meleknazablak@gmail.com

# Özet

Bizden C dillini kullanarak Coğrafi Bilgi Sistemi (CBS), dünya üzerindeki karmaşık sosyal, ekonomik, çevresel vb. sorunların çözümüne yönelik mekâna/konuma dayalı karar verme süreçlerinde kullanıcılara yardımcı olmak üzere, büyük hacimli coğrafi verilerin; toplanması, depolanması, işlenmesi, yönetimi, mekânsal analizi, sorgulaması ve sunulması fonksiyonlarını yerine getiren donanım, yazılım, personel, coğrafi veri ve yöntem bütünüdür. Bu proje kapsamında, şehirlerin komşuluk ilişkilerini dikkate alan çok basit bir harita verisi üzerinde bağlı listelerle ilgili temel işlevleri yerine getiren bir uygulama yapılması istendi. Tasarladığınız sistem verilen özelliklere sahip sınırsız sayıda şehir bilgisini ve komşuluk bilgisini tutabilmelidir. Dinamik bir yapıda olmalıdır.Tasarlanan sistem formatı açıklanan “sehirler.txt” dosyasındaki kayıtları okuyarak ve düğüm yapısını kullanarak modele aktarmalıdır. Kullanıcı yeni bir şehir, yeni bir komşuluk ekleyebilmelidir. Fakat aynı isimli şehir ve özellikte yeni bir kayıt eklenmesine sistem izin vermemelidir.

# 1.Giriş

Proje konusu; C dili ile dosyalama işlemleri ve bağlı liste kullanılarak Coğrafi Bilgi Sistemi (CBS) programının oluşturulması.

Projenin amacı, C programlama dilini ve dosyalama işlemlerini kullanarak çift yönlü bağlı liste ve tek yönlü bağlı listede ekleme, çıkarma, başa, araya, sona sıralı ekleme işlemlerinin kullanılması.

# 2.Temel Bilgiler

Proje C dilinde yazıldı ve CodeBlocks’da hazırlandı.

 **CodeBlocks**, Code:Blocks özellikle C ve C++ gibi programlama dillerinde kodlama geliştirme yapmamıza imkan veren, açık kaynak kodlu ve cross-platform bir IDE’dir (Integrated Development Environment).  
Code:Blocks IDE ortamının kendisi de C++ ile geliştirilmiştir.  wxWidgets tabanlı tamamen özelleştirilebilir arabirimiyle, GNU/Linux, Microsoft Windows, MacOS platformlarında sorunsuzca kullanılabilmektedir.

##  **Notepad++**,

## Notepad++, Windows işletim

## sistemi içerisine gömülü

## olarak gelen Notpead yazılımının

## yerine kullanılmak üzere C++

## ile saf Win32 API ve STL ile geliştirilmiş

## GPL ile dağıtılan açık kaynak

## kodlu bir kaynak kod düzenleyicisidir.

## 

# 3.Diğer Bölümler

Bu modelde her bir şehrin dört tane özelliğe sahip olduğu bilinmektedir: şehrin ismi, plaka kodu, hangi bölgede olduğu ve bu şehirden direk ulaşılabilen (komşu) şehirler. Şehirler arasındaki komşuluk ilişkisi bir pointer (işaretçi) yardımıyla tutulabilir. Artık her bir şehir bir bağlı listenin düğümleri şeklinde tutulabilir.



## 3.1 Şehri Dosyadan Oku

Fgets() fonksiyonu ile dosyadan satır satır okuma işemi yaparak ayraç olarak tanımlaadığımız değişken ile virgül sayısına göre kelimeleri ayırdık. Tanımladığımız değişkenler olan plaka, sehir\_adi, bölge\_adi, komsu\_sayisi değişkenlerine dosyadan okuduğumuz karşılıkarını atadık.

### 3.2. Dosyadan Şehir Ekle

Bu fonksiyonda dosyada komşu şehirlerin verilmeyen plakalarının şehirlerde verilen

Plakalardan şehirlerin plakalarını tutarak sağladık.

### 3.3. Kullanıcı Şehir Ekle

Bu fonksiyonda kullanıcının girmek istediği

şehrin plakasını istiyoruz.Girilen plakayı ara()

fonksiyonuna yollayarak şehrin okuduğumuz

dosyada olup olmadığının kontrolünü yapıyoruz.

Eğer şehir varsa “eklemek istediginiz sehir zaten var..” mesajı ile olan şehrin tekrardan eklenmesine

engel oluyoruz. Ara fonksiyonu null dönerse şehrin

oladığı ve kullanıcının eklemesi için malloc() ile yer ayırıyoruz ve kullanıcıdan şehrin ; plaka, sehir\_adi, bölge\_adi, komsu\_sayisi bilgilerini alıyoruz.

### 3.2. Sil Şehir

Bu fonksiyonu struct tipinde tanımladık.

Fonksiyonun döndürdüğü değer kullanıcını silmek

istediği şehrin adresini döndürmektedir.

### 3.3. Komşu Dosyadan Oku

Bu fonksiyonda fgets() fonksiyonu ile

dosyadan satır satır okuma işemi yaparak

ayraç olarak tanımlaadığımız değişken

ile virgül sayısına göre kelimeleri ayırdık.

Şehirlerin tuttuğumuz plakalarına göre

komşu şehirlerin karşılık gelen plakalarıyla

ekliyoruz.

### 3.4. Kullanıcı Komşu Şehir Ekle

Bu fonksiyonda kullanıcının yaptığı

seçim sonucu ekrana “komsu

olarak eklemek istediginiz sehrin adini

giriniz:" mesajını veriyoruz .Malloc()

fonksiyonu ile eklenmek istenen

komşu şehre yer ayırıyoruz. Kullanıcının

girdiği komşu şehir adını şehir\_ara()

fonksiyonuna gönderiyoruz. Eğer

eklemek istenilen komşu şehir şehirlerde

yok ise kullanıcıyı kullanici\_sehir\_ekle()

fonksiyonu ile şehir eklemeye yönlendiriyoruz.

Eğer eklemek istenilen komşu şehir şehirlerde

var ise “hangi sehire komsuluk eklemek

istersiniz" mesajı ile komşuluk eklenilecek şehri

kullanıcıdan istiyoruz.Komşuluk eklenilecek

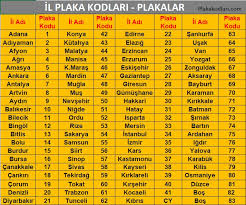
şehrin komşu olarak önceden var olup

olmadığının kontrolünü yapıyoruz.

Komşu şehir var ise kullanıcın eklemesine

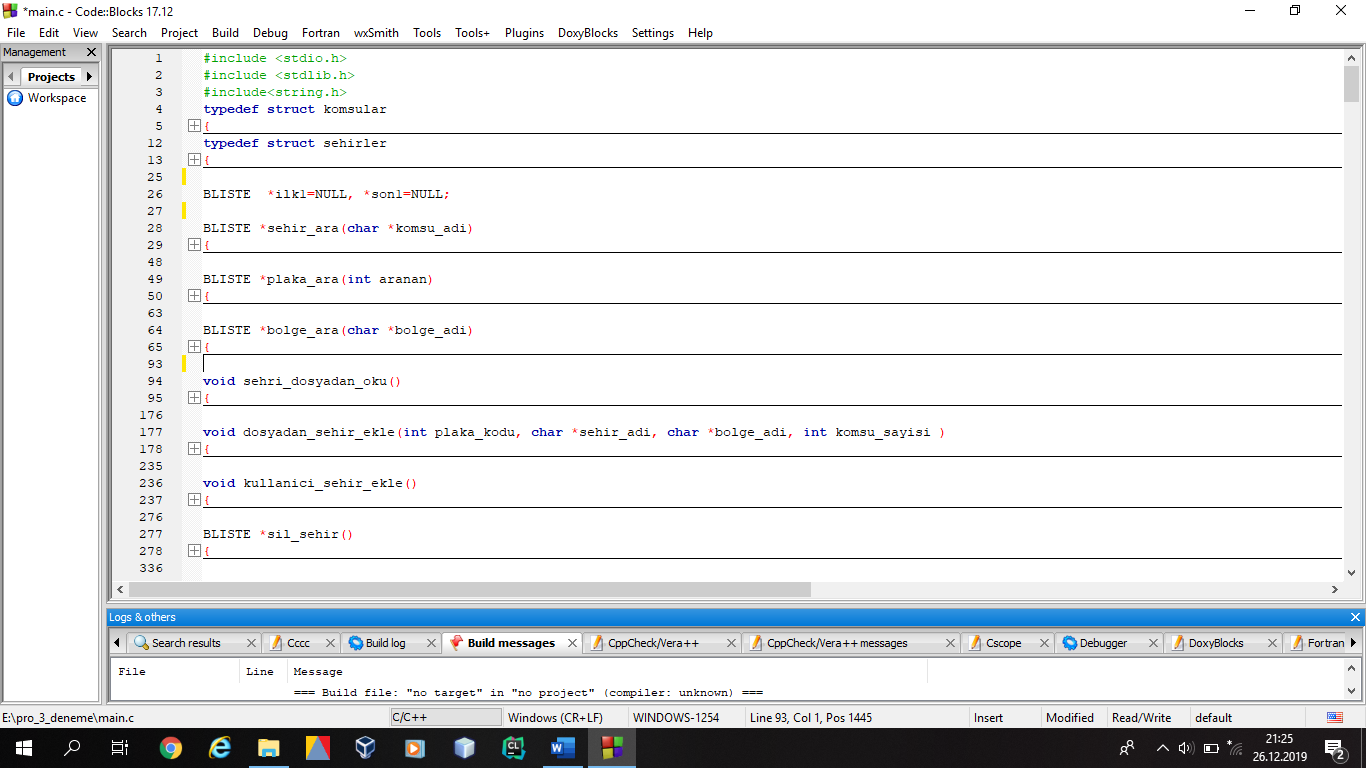
engel oluyoruz, yok ise komşuluk

plaka sırasına göre sıralı ekleme ile ekleniyor.



### 3.5. Sil Komşu

Bu fonksiyonu struct tipinde struct komsular sil\_komsu(); olarak tanımladık. Fonksiyonu struck tipinde tanımlamamızın nedeni döndürdüğü değerin kullanıcını silmek istediği komşu şehrin adresini döndürmektedir. Kullanıcıdan hangi şehirden komşuluk silmek istediği soruluyor. Girilen isim listede yoksa “mevcut değil” mesajı veriliyor. Var olduğunda ise plaka ara fonksiyonundan o şehrin plakası tutuluyor ve listeden siliniyor.



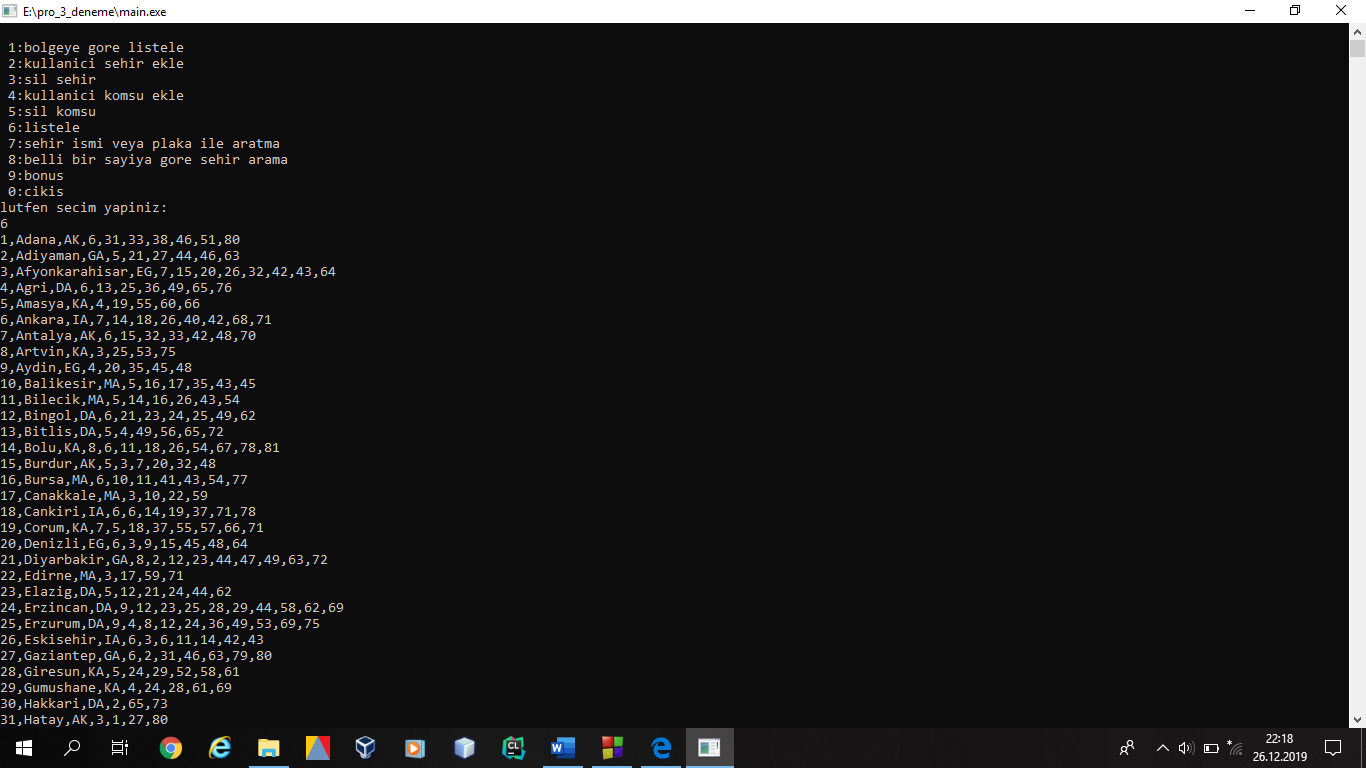
### 3.6. Listele

Bu fonksiyonda dosyadan satır satır okuduğumuz şehirlerin plaka kodlarını, şehir adlarını, böl adlarını, komşu sayılarını ve komşu şehirlerini plaka kodlarına göre sıralı eklediğimiz bağlı listeden görüntülüyoruz.

**4.Sonuçlar**

Tasarlanan bağlı liste uygulamasında sisteme ilk başta belirtilen formatı açıklanan “sehirler.txt” dosyasındaki kayıtları okuyarak ve düğüm yapısını kullanarak modele aktarır.

Tasarladığımız sistem istenilen özelliklere sahip sınırsız sayıda şehir bilgisini ve komşuluk bilgisini tutabilmektedir. Dinamik bir yapıdadır. .Kullanıcı herhangi bir bölgede bulunan şehirlerin bilgilerini (plaka kodu, şehir adı, komşu sayısı) listeleyebilir. Belli bir komşu sayısı kriterine uyan şehirler bulunabilir ve gösterebilir. Kullanıcı bir şehri veya komşuluk kaydını silebilir. (Olmayan kayıtların silinmesine sistem izin verilmez.) Kullanıcı yeni bir şehir, yeni bir komşuluk ekleyebilir. Fakat aynı isimli şehir ve özellikte yeni bir kayıt eklenmesine sistem izin vermez.



**5.Pseudo Code(Kaba Kod)**

Bu kısımda programımızın şehir ara, kullanıcı şehir ekle ve kullanıcı komşu sil fonksiyonlarının kaba kodu bulunmaktadır.

1. BLISTE \*sehir\_ara(char \*komsu\_adi);

1.Başla

2.Şehir adı komşu adına eşit olanakadar while nin içinde dön,

3.Eğer şehir adı komşu adına eşitse ismi döndür,

4.Eğer şehir adı komşu adına eşit değilse NULL döndür,

5.Bitir.

1. void kullanici\_sehir\_ekle();

1.Başla

2.İnteger tipinde plaka tanımla,

3.Kullanıcıdan değer al,

4.Değer plaka ara fonksiyonundan NULL döndüğünde şehrin bilgilerini al,

5.Dosyadan şehir ekle fonksiyonuna plaka şehir adı , bölge adı, komşu sayısı gönderildi,

6. Eğer değer plaka ara fonksiyonundan NULL döndürmediğinde ise “eklemek istediğiniz şehir zaten var” mesajı görüntüle ,

7.Bitir

1. BLISTE \*sil\_sehir();

1.Başla,

2.Silmek istenen şehrin plakası kullanıcı tarafından al,

3.Plaka ara fonksiyonuna gönder,

4.Eğer NULL dönerse “böyle bir şehir mevcut değildir” mesajını görüntüle,

5.Eğer NULL dönmezse eğer ilk şehrin plakasından küçükse baştan sil,

6.Değilse eğer sondan büyükse sondan sil,

7.Değilse aradan sil,

8.Bitir.

1. void sehri\_dosyadan\_oku();

1.Başla,

2.sehirler.txt okuma(r) modunda aç,

3.fgets() fonksiyonu ile bir satır al,

4.strtok() fonksiyonu ile ayracı virgül belirle,

5.Kelime NULL değilse ve sayaç 4 den küçük olduğu sürece dön,

6.Alınan kelimeleri uygun değişkenlere ata,

7.Aldığımız değişkenleri dosyadan şehir ekle fonksiyonuna sırası ile plaka , şehir adı, bölge adı, komşu sayısı şeklinde gönder,

8.Dosyayı kapat,

9.Bitir.

**6.Kaynakça**

[1]. Web Site

<https://www.programiz.com/c-programming/c-arrays-functions>

[2]. Web Site

<https://www.tutorialspoint.com/cprogramming/c_functions.htm>

[3]. Web Site

<https://www.geeksforgeeks.org/cprogramming/fileoperations>

[4]. Web Site

<https://www.fresh2refresh.com/cprogramming/fileoperations>

[5]. Web Site <https://www.geeksforgeeks.org/basics-file-handling-c/>

[6]. Web Site

<https://drive.google.com/drive/folders/1EF1e6nB3I4Z_PEFCYKMCQ4mzkeqmdFDE>

# 