****

**MARMARA ÜNİVERSİTESİ TEKNOLOJİ FAKÜLTESİ**

**BİLGİSAYAR MÜHENDİSLİĞİ**

Ahmet Faruk Alkan 170420044

Mustafa Eren Gülbahar 170420019

İsmail Can Varlı 171420004

**Proje Metodu**

Projemiz bir "Emlak Otomasyonu". Projede yapmak istediklerimiz konut bilgisini girme ve kaydetme, görüntüleme, konut ekleme,silme ve güncelleme, konutları belli parametrelere göre misal olarak konut fiyatı, oda sayısı, konut kat bilgisisi, m2 bilgisi, konum bilgisi vs. sıralama ve filtreleme işlemleri olacak.

Bu işlemler için bir C derleyici kullandık. Yazdığımız kod C dili için derlenmeli yoksa hata verebilir.

Otomasyonumuzda ki konut bilgilerini 500 elemanlı bir struct dizisine attık. Konut bilgileri için değişkenlerimiz konut alanı, konut kat bilgisi, konut durumu, konut fiyatı, arsa bilgisi, il bilgisi, ilçe bilgisi ve oda sayısı. Default değer olarak "char" veri tipleri için NULL, "integer" veri tipleri için 0 değerlerini verdik. Ana bölüm için bir panel oluşturduk ve bu paneli "do-while" döngüsüne attık. Bu panelde seçimlerimize göre bazı işlemler yapacak. Eğer seçim olarak 0 girerseniz kodumuz kapanır. Bu ana panelde konut görüntüleme, ekleme, silme, güncelleme ve filtreleme seçenekleri var. İşlemlere daha yakından bakalım.

Konut ekleme işlemini seçersek burada kaç konut gireceğimizi sorar ve girdiğimiz değer kadar konut bilgilerini gireriz. Burada "integer" değerleri scanf ile, "char" veri tipindeki değerleri "gets" fonksiyonu ile alıyoruz. Girdiğimiz verileri struct dizisine kaydederiz.

Konut görüntüleme fonksiyonunu seçersek struct dizisine kaydettiğimiz konut bilgilerini bir bar ile birlikte görüntüleriz.

Konut sil fonksiyonunu seçersek burada ilk olarak konut görüntüle fonksiyonunu çalıştırır. Ve buradan sileceğimiz konutun konut numarasını girerek o numaradaki konut bilgilerini struct dizisi üzerinden "char" veri tipleri için NULL, "integer" veri tipleri için 0 değerini atarız. Silme işlemi sonucunda struct dizisindeki ardışıklık bozulabilir. Demek istediğim 2. sıradaki verileri sildiğimde 3. sıradaki veriler 2. sıraya kaymalı. Bunun için ise "konutkaydir" ve "konutduzenle" adında iki fonksiyon yazdık. Eğer struct dizisi içinde bir boşluk oluşursa bu boşluğu bulup struct dizisini o boşluğa göre kaydırmak için bu iki fonksiyonu kullandık.

Konut güncelleme fonksiyonunu seçersek ilk olarak konut görüntüle fonksiyonunu çalıştırılır ve burada seçtiğiniz konut numarasına göre o numaradaki konut bilgilerini tekrardan gireriz.

Tüm bu işlemleri struct dizisi içinde yaptık ama bu verileri kalıcı hale getirmek için tüm bu verileri bir txt dosyası içine yazdık. Kodu ilk çalıştırdığımızda eğer bir txt dosyası yoksa bir "KONUT.txt" adında txt dosyası oluşturulur. Eğer bir txt dosyası varsa buradan verileri "fscanf" ile struct dizisine doldurduk. Bu işlem için "txtoku" fonksiyonunu yazdık. Daha sonra yukarıda bahsettiğim ekleme, silme, güncelleme, işlemleri sonucunda bu veriler değişebilir. Bu değişiklikleri kaydetmek için ise "do-while" fonksiyonunun sonuna txt dosyasına da bu değişiklikleri kaydetmek için "txtyaz" fonksiyonunu yazdık. Burada struct verilerini txt dosyasına bastırdık.

**Çoklu Filtreleme**

Bu filtreleme işleminde birden fazla parametre ile yapılır bu parametreleri kullanıcı girer. Girilen parametrelere göre kullanıcıdan bazı veriler istenir. Fiyat filtrelemesi için maksimum minimum fiyat,kat filtrelemesi için maksimum minimum kat, m2 filtrelemesi için maksimum minimum m2, arsa durumu için satılık-kira, oda sayısı bilgisi vs. Girilen bu verilere göre struct dizisindeki bilgiler filtrelenir ve kullanıcıya sunulur.

**Oda Sayısı**

Kullanıcıdan arayacağı oda sayısını girmesini isteyeceğiz. Girdiği oda sayısını struct dizisindeki oda sayısı bilgisi ile "strcmp" fonksiyonu yardımıyla karşılaştıracağız. Bu işlemi konut bilgilerini bastırmadan şartlandıracağız ve kullanıcının girdiği oda sayısı ile aynı olanlar bastırılacak.

**Metrekare Filtreleme**

Metrekare filtrelemede kullanıcıdan bir seçim yapması için 3 seçenek sunuyoruz. Birinci seçenekte konutların fiyatlarını küçükten büyüğe sıralamak istersek kullanıyoruz. Sıralamak için kullandığımzı algoritma şu şekilde : Girdiğimiz konutların indisini ve değerini bir diziye matris şeklinde atıyoruz. Ardından küçükten büyüğe sıralamak için kıyas yapıyoruz. Bu kıyası yaparken indislerin de yerini değiştiriyoruz, böylece indisleri tuttuğumuz diziyi asıl dizimizin içine yazarak bastırıyoruz. Bu algoritmayı büyükten küçüğe sıralama yaparken de kullanıyoruz. Bu sıralama işlemini yaparken başlangıçta girdiğimiz konut sırasını değiştirmemiş oluyoruz. Filtreleme içinde sıralama değişiyor sadece. Yani filtreleme esnasında sıralaması değişiyor. İkinci seçenekte büyükten küçüğe doğru sıralama yapıyoruz. Bu sıralamayı yaparken başlangıçta girdiğimiz konut sırasını bozmuyoruz. Filtreleme içerisinde konutların yerini değiştiriyoruz. Üçüncü ve son seçenek ise çıkış yapmak istersek kullanıyoruz.

**Kat filtreleme**

Kat filtrelemede kullanıcıdan bir seçim yapması için 3 seçenek sunuyoruz. Birinci seçenekte konutların kat sayısının küçükten büyüğe sıralıyoruz. Sıralamak için kullandığımzı algoritma şu şekilde : Girdiğimiz konutların indisini ve değerini bir diziye matris şeklinde atıyoruz. Ardından küçükten büyüğe sıralamak için kıyas yapıyoruz. Bu kıyası yaparken indislerin de yerini değiştiriyoruz, böylece indisleri tuttuğumuz diziyi asıl dizimizin içine yazarak bastırıyoruz. Bu algoritmayı büyükten küçüğe sıralama yaparken de kullanıyoruz. Bu sıralamayı yaparken başlangıçta girdiğimiz konutların sıralaması bozulmuyor. Filtreleme içerisinde sıralamayı değiştiriyoruz. Yani filtreleme esnasında sıralaması değişiyor. İkinci seçenekte büyükten küçüğe sıralıyoruz. Bu sıralamayı yaparken başlangıçta girdiğimiz sıralama bozulmuyor, filtreleme içerisinde değişiyor sadece. Üçüncü ve son seçenek ise çıkış yapmak istersek kullanıyoruz.

**Konum bilgisi**

Filtreleme olarak kullanılacak olan daire veya mülkün konum bilgisinde amaçladığımız şudur. Daire ya da mülkün il/ilçe bazında nerede bulunduğudur. Önce il bilgisi sonra ilçe bilgisi girilerek daha spesifik bir filtreleme gerçekleştirmiş oluyoruz. Bunun için öncelikle programı çalıştırmamız gerekli ondan sonra açılan ana menüde filtreleme yapabilmek için “5” rakamını giriyoruz ve “enter”e basıyoruz. Menüde de yazdığı gibi konum filtrelemesi yapmak için “7” rakamını yazıyoruz. Ondan sonra il, ilçe veya ikisinin de içinde bulunduğu filtreleme seçeneklerinden birini seçiyoruz. Bu aşamadan sonra seçtiğimiz seçeneğe bağlı olarak il ilçe bilgilerini giriyoruz ve sisteme kayıtlı daire veya mülk varsa bunların bilgilerine ulaşmış oluyoruz.

Eğer filtreleme yapma kısmından çıkmak istiyorsak “0” rakamına basıp “enter”e basıyoruz ve bir önceki menüye dönüyoruz.

**Arsa bilgisi**

Bu filtrelemedeki maksadımız şudur. Arsanın durumunu öğrenebilmek (apartman, dükkan, ya da müstakil) şeklinde. Ana menüde “5” rakımına basıp “enter” tuşuna basıyoruz. Açılan menüde “6” tuşuna basıyoruz ve devamında “enter” tuşuna basıyoruz. Menüde de gözüktüğü gibi filtreleme yapmak istediğimiz seçeneği yazıp “enter” tuşuna basıyoruz ve filtrelemeye uygun seçenekler karşımıza çıkıyor.

**Fiyat bilgisi**

Bu filtrelemede kullanıcıya sunulacak olan şudur. Dairelerin artan, azalan veya belirli fiyat aralığında sıralayabilme seçeneğini sunar. Önce ana menüde “5” tuşuna basıp sonra “enter” tuşuna basıyoruz ve açılan menüde “2” tuşuna basıp “enter” tuşuna basıyoruz. Fiyatları artan sıra ile görüntülemek istiyorsak “1” , azalan sıra ile görüntülemek istiyorsak “2” ,fiyat aralığı belirleyip bu aralığa göre görüntülemek istiyorsak “3” tuşuna basıp “enter” tuşuna basıyoruz. Fiyat aralığı belirlerken önce minimum fiyat sonra maksimum fiyatları giriyoruz ve uygun seçenekler karşıma çıkıyor.

**Arsa durumu**

Bu filtreleme seçeneğinde arsanın “Satılık mı? kiralık mı? ” olduğunu filtreliyoruz. Bunun için açılan menüde önce “5” tuşuna basıyoruz ve “enter” tuşuna basıyoruz. Ondan sonra açılan menüde “5” tuşuna basıyoruz ve tekrar “enter” tuşuna basıyoruz. Eğer baktığımız daire yada arsalar satılık ise “1” , kiralık ise “2” tuşuna basıyoruz. Ve filtreleme sonucu uygun seçenekler karşımıza çıkmaktadır.