

**İSTANBUL SAĞLIK VE TEKNOLOJİ ÜNİVERSİTESİ**

Mühendislik Fakültesi

Yazılım Mühendisliği Bölümü

BYM102 – İşletim Sistemleri

**DİYETİSYEN RANDEVU SİSTEMİ**

Hazırlayan:

Ad Soyad : ONUR ŞAHİN  
Öğrenci Numarası : 230609029  
Sınıf : 2.SINIF

Dersi Veren:

MEHMET GÜÇLÜ

Teslim Tarihi:

30.05.2025

**Teknik Rapor: Akıllı Randevu Planlayıcı ve Takip Sistemi**

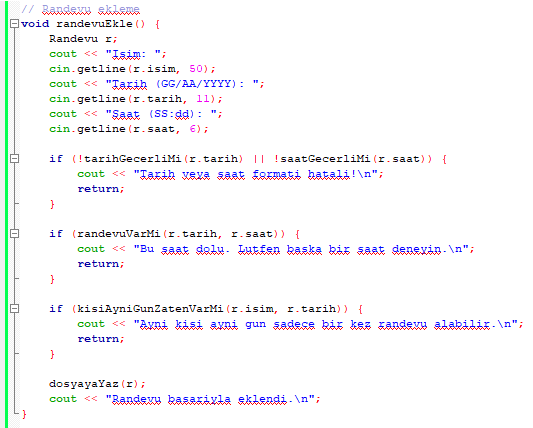
**1. Giriş**

Bu proje, kullanıcıların belirli tarih ve saatlerde çakışma olmaksızın randevu almasını, silmesini, güncellemesini ve listelemesini sağlayan bir C++ uygulamasıdır. Kullanıcı arayüzü konsol üzerinden sağlanmakta ve veriler randevular.txt ile silinmisler.txt dosyalarında saklanmaktadır.

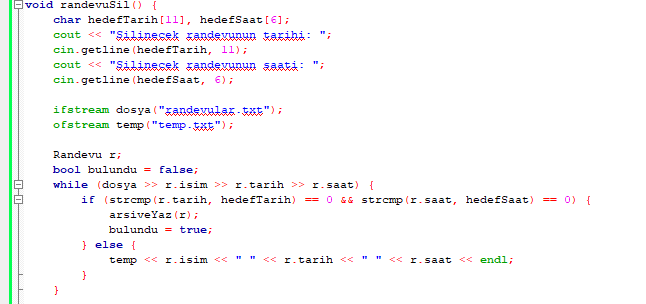
**2. Kullanılan Programlama Yapıları**

**Fonksiyonlar**

Kodda işlevselliği bölmek ve tekrarlanan işlemleri soyutlamak amacıyla çok sayıda fonksiyon kullanılmıştır:



* void randevuEkle(): Kullanıcıdan randevu bilgilerini alır, doğrulama yapar ve dosyaya ekler.



* void randevuSil(): Belirli tarih ve saate ait randevuyu bulup siler ve silinen randevuyu arşive yazar.

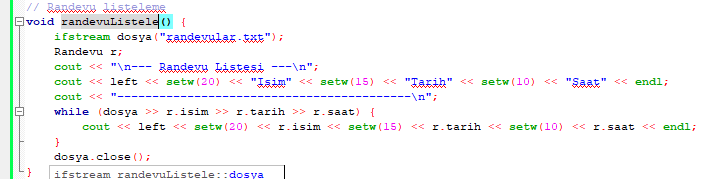
metin, ekran görüntüsü, yazı tipi, sayı, numara içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.

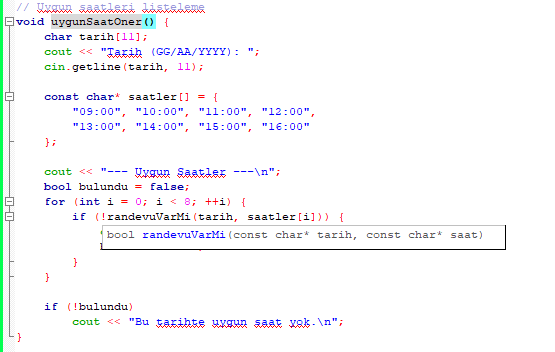
metin, ekran görüntüsü, yazı tipi, çizgi içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.

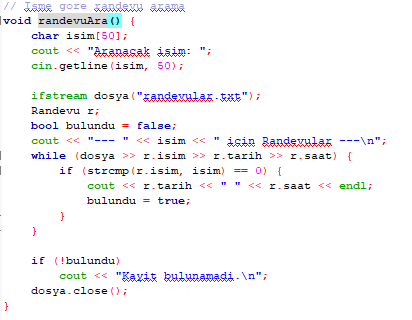
* void randevuGuncelle(): Var olan bir randevunun bilgilerini değiştirir.



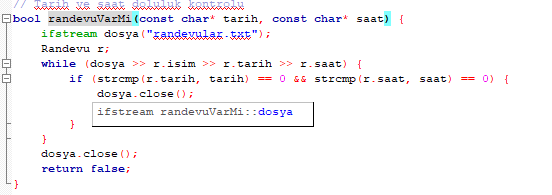
* void randevuListele(): Tüm kayıtlı randevuları listeler.



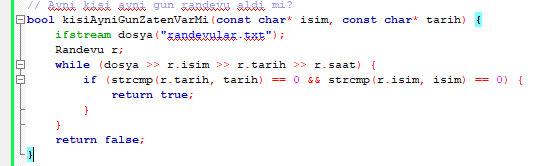
* void uygunSaatOner(): Belirli bir tarihteki uygun saatleri önerir.



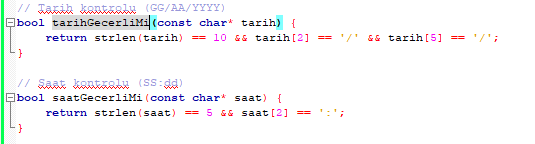
* void randevuAra(): İsimle arama yapar.



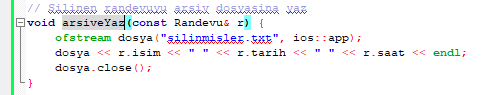
* bool randevuVarMi(...): Aynı tarih ve saat kombinasyonunun kullanılıp kullanılmadığını kontrol eder.



* bool kisiAyniGunZatenVarMi(...): Aynı kişinin aynı gün birden fazla randevusu olup olmadığını kontrol eder.



* bool tarihGecerliMi(...): Tarih formatının doğruluğunu kontrol eder.
* bool saatGecerliMi(...): Saat formatının doğruluğunu kontrol eder.



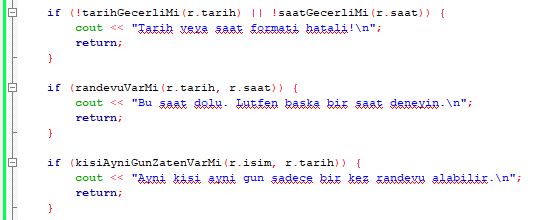
* void arsiveYaz(...): Silinen randevuyu arşiv dosyasına ekler.

**Fonksiyonel programlama** ilkeleri gereği her işlem kendi içerisinde bağımsız olarak ele alınmıştır.

**Karar Yapıları (Koşullar ve switch-case)**

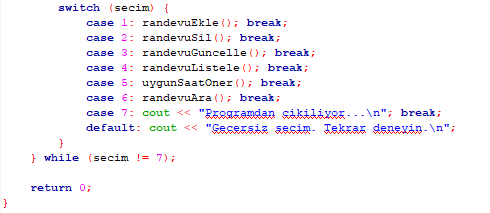
* if-else yapıları, tarih ve saat doğrulaması, randevunun uygunluğu ve kullanıcı girişlerinin kontrolünde sıkça kullanılmıştır.

Örnek:



* switch-case yapısı ana menü kontrolünde tercih edilmiştir. Kullanıcının menüden yaptığı seçime göre ilgili fonksiyon çağrılır:

Örnek:



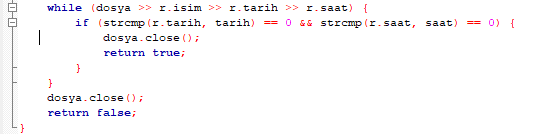
* return ifadesi, hatalı girişlerde veya işlemi bitirme gerekliliğinde fonksiyondan çıkmak için kullanılmıştır.

**Döngüler ve İç İçe Döngüler**

**Temel Döngüler**

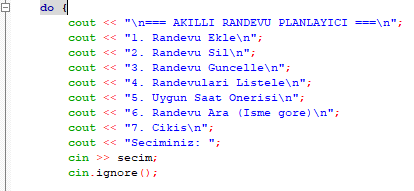
* while döngüsü dosya içeriğini satır satır okumak için kullanılmıştır.

Örnek:



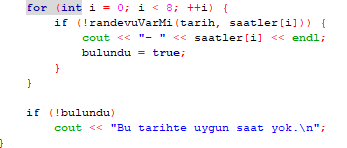
**do-while Döngüsü**

* Ana menü sürekli kullanıcıdan giriş almak için do-while ile yazılmıştır:



**for Döngüsü**

* Uygun saat önerisinde sabit bir dizi üzerinden tekrar yapmak için kullanılmıştır:

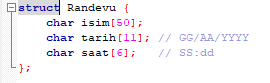


**Kontrol İfadeleri**

* **break:** Kullanıcı çıkış seçeneğini seçtiğinde döngüden çıkmak için kullanılmıştır.
* **continue:** Kullanıcıdan alınan bilgiler hatalıysa işlemi sonlandırmak için genelde return tercih edilmiştir.

**Karakter Dizileri (C-style Strings)**

Karakter dizileri kullanılmıştır çünkü struct Randevu sabit boyutlu char[] alanlarına sahiptir:



* Kullanıcı girişleri cin.getline(...) ile alınmıştır.
* Karşılaştırmalar için strcmp(...) fonksiyonu kullanılmıştır.
* Format doğrulamaları için strlen(...) ve karakter indeksleri kullanılmıştır.

**İşaretçiler (Pointers)**

* Fonksiyonlara const char\* şeklinde parametre verilerek işaretçilerle çalışılmıştır:



Bu sayede diziler doğrudan kopyalanmadan, referansla okunur. Bellek verimli kullanılmış olur.

**Dosya İşlemleri**

**Dosya Okuma (ifstream)**

* Randevular randevular.txt dosyasından satır satır okunur.
* ifstream ile okuma yapılır, her satır Randevu yapısına aktarılır.



**Dosya Yazma (ofstream)**

* Yeni randevular veya arşivlenenler için ofstream ile dosya sonuna yazma (ios::app) yapılır:



**Geçici Dosya Kullanımı**

* Randevu silme veya güncelleme işleminde temp.txt dosyası kullanılarak yeni içerik oluşturulur.
* Ardından eski dosya silinip yenisi yeniden adlandırılır:



**EKRAN ÇIKTILARI**

**metin, ekran görüntüsü, yazı tipi içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.metin, ekran görüntüsü, yazı tipi, sayı, numara içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.metin, yazı tipi, ekran görüntüsü, tipografi içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.metin, ekran görüntüsü, yazı tipi içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.**

**ALGORİTMA**

**1. Başlangıç ve Temel Yapı**

1. Gerekli kütüphaneleri ve başlık dosyalarını ekle (iostream, fstream, cstring, iomanip, vector, ctime, sstream, algorithm)
2. Sabitleri tanımla:
   * Randevu dosyası adı
   * Arşiv dosyası adı
   * Standart saatler listesi
3. Randevu struct'ını oluştur:
   * İsim, tarih ve saat alanları
   * Ekrana yazdırma fonksiyonu

**2. Yardımcı Fonksiyonlar**

1. dosyaVarMi() - Dosyanın varlığını kontrol et
2. randevulariOku() - Dosyadan randevuları oku ve vektöre yükle
3. randevulariYaz() - Randevu vektörünü dosyaya yaz
4. tarihGecerliMi() - GG/AA/YYYY format kontrolü yap
5. saatGecerliMi() - SS:dd format kontrolü yap
6. randevuVarMi() - Belirtilen tarih/saatte randevu olup olmadığını kontrol et
7. kisiAyniGunZatenVarMi() - Aynı kişinin aynı gün randevu alıp almadığını kontrol et
8. arsiveEkle() - Silinen randevuları arşiv dosyasına ekle
9. bugununTarihi() - Sistem tarihini GG/AA/YYYY formatında döndür

**3. Ana Fonksiyonlar**

1. **Randevu Ekleme (randevuEkle)**
   1. Kullanıcıdan isim, tarih ve saat bilgilerini al
   2. Tarih ve saat formatlarını kontrol et
   3. Randevu listesini oku
   4. Aynı tarih/saatte randevu olup olmadığını kontrol et
   5. Aynı kişinin aynı gün randevu alıp almadığını kontrol et
   6. Tüm kontroller geçilirse randevuyu listeye ekle ve dosyaya yaz
2. **Randevu Listeleme (randevuListele)**
   1. Tüm randevuları dosyadan oku
   2. Formatlı şekilde ekrana yazdır
3. **Bugünün Randevuları (bugununRandevulari)**
   1. Sistem tarihini al
   2. Bugüne ait randevuları filtrele
   3. Formatlı şekilde ekrana yazdır
4. **Randevu Silme (randevuSil)**
   1. Kullanıcıdan silinecek randevunun tarih/saat bilgisini al
   2. Randevu listesini oku
   3. Belirtilen randevuyu bul ve arşiv dosyasına ekle
   4. Güncellenmiş listeyi dosyaya yaz
5. **Randevu Güncelleme (randevuGuncelle)**
   1. Kullanıcıdan güncellenecek randevunun eski tarih/saat bilgisini al
   2. Randevu listesini oku
   3. Belirtilen randevuyu bul
   4. Yeni bilgileri al ve kontrollerden geçir
   5. Güncellenmiş listeyi dosyaya yaz
6. **Uygun Saat Önerisi (uygunSaatOner)**
   1. Kullanıcıdan tarih bilgisini al
   2. Tarih formatını kontrol et
   3. Standart saatler listesinde boş olanları bul
   4. Uygun saatleri listele
7. **Randevu Arama (randevuAra)**
   1. Kullanıcıdan aranacak ismi al
   2. Randevu listesinde isme göre filtreleme yap
   3. Bulunan randevuları listele

**4. Ana Program Akışı (main)**

1. Randevu dosyasını kontrol et, yoksa oluştur
2. Kullanıcıya menüyü göster
3. Kullanıcı seçimine göre ilgili fonksiyonu çağır
4. Çıkış seçilene kadar döngüyü sürdür

**5. Veri Yapıları ve Akış**

1. Randevu bilgileri struct ile tutulur
2. Tüm randevular bir vektörde saklanır
3. Dosya işlemleri için metin dosyaları kullanılır
4. Tarih/saat validasyonları basit kontrollerle yapılır

**6. Hata Kontrolleri**

1. Dosya varlığı kontrolü
2. Tarih/saat format kontrolleri
3. Randevu çakışması kontrolleri
4. Aynı gün aynı kişi kontrolü
5. Randevu bulunamadı durumları

**AKIŞ ŞEMASI**

metin, diyagram, ekran görüntüsü, plan içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.

ALTERNATİF :



ALTERNATİF 2 :  [https://drive.google.com/file/d/1pwIL4d9D1\_UqSdKPQwo6cwppzvmf0-iQ/view?usp=sharing](%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20https://drive.google.com/file/d/1pwIL4d9D1_UqSdKPQwo6cwppzvmf0-iQ/view?usp=sharing)