yazı tipi, metin, tipografi, hat sanatı, kaligrafi içeren bir resim

Açıklama otomatik olarak oluşturuldu

**2024-2025 BAHAR DÖNEMİ**

**MÜHENDİSLİK VE DOĞA BİLİMLERİ FAKÜLTESİ**

**YAZILIM MÜHENDİSLİĞİ**

**BYM 102**

**BİLGİSAYAR PROGRAMLAMA II**

**PROJE RAPORU**

**KÜTÜPHANE TAKİP SİSTEMİ**

**ÖDEV TESLİM TARİHİ**

**30.05.2025**

**HAZIRLAYAN**

**ONUR ŞAHIN - 230609029**

**DERSİN EĞİTMENİ**

**Asst. Prof. Dr. Mehmet Güçlü**

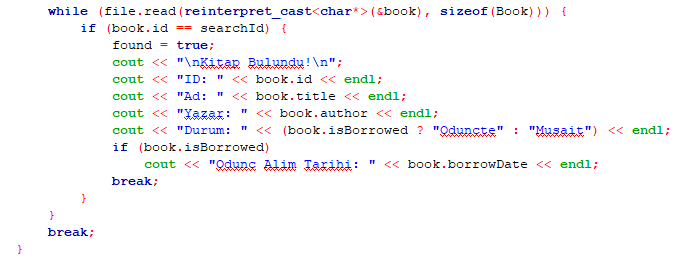
**Projemde amacım bir kütüphanedeki kayıtlı bulunan kitapları sistemde belirli özelliklerle tutmak ve ödünç alınıp verilme sisteminin düzgün bir şekilde takip edilmesini sağlamaktı. Bizden kullanmamız istenen;**

* **Döngü kontrol ifadeleri (break, continue, goto ve return)**
* **İç içe döngüler**
* **Karar yapıları / ifadeleri (switch-case yapısı mutlaka işlenmelidir)**
* **Diziler (Karakter dizileri mutlaka işlenmelidir)**
* **Fonksiyonlar**
* **İşaretçiler**
* **Dosya işlemleri**

**konularını eksiksiz ele alarak kafamda planladığım projeyi uygulamaya geçirdim. Raporumda sizlere uygulamanın koduna ve kod çalıştığında meydana gelecek olağan sonuçlara dair ekran görüntülerini ekleyip, bazı kod bloklarının ayrıyeten görsellerini ve açıklamalarını sunacağım. Programın algoritmasını ve akış diyagramını da ekleyerek çalışma mantığını incelemenizi sağlayacağım.**

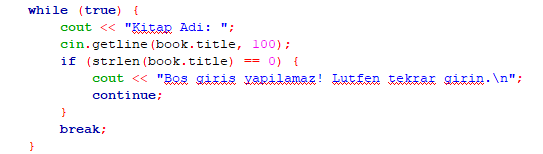
**1. Döngü Kontrol İfadeleri**

**break Kullanımı**

****

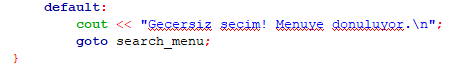
**break ifadesi döngüyü anında sonlandırır ve kontrolü döngüden sonraki ilk ifadeye aktarır.**

**continue Kullanımı**

****

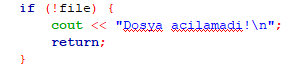
**continue ifadesi mevcut döngü iterasyonunu atlar ve bir sonraki iterasyona geçer.**

**goto Kullanımı**

****

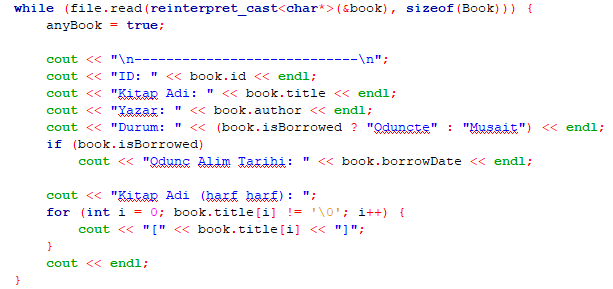
**goto belirtilen etikete atlamak için kullanılır (genellikle kaçınılması önerilir).**

**return Kullanımı**

****

**return fonksiyonun çalışmasını sonlandırır ve çağrıldığı yere döner.**

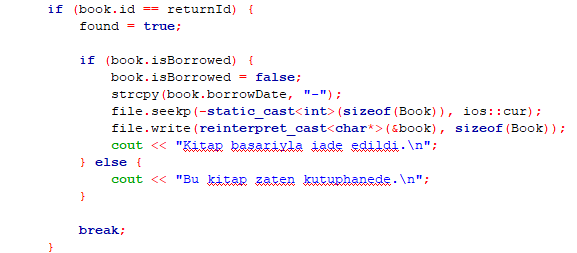
**2. İç İçe Döngüler**

****

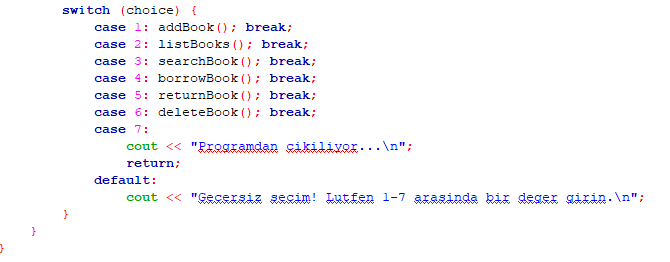
**Bir while döngüsü içinde for döngüsü kullanılmıştır.**

**3. Karar Yapıları**

**if-else Yapısı**

****

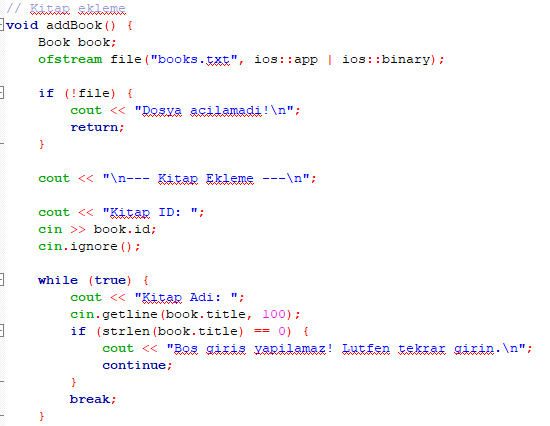
**switch-case Yapısı**

****

**4. Fonksiyonlar**

**void addBook()**

**Amaç: Yeni bir kitabı books.txt dosyasına eklemek.**

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

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.**

**Adımlar:**

1. **Book yapısı oluşturulur ve kullanıcıdan ID, başlık (title), yazar (author) bilgileri alınır.**
2. **isBorrowed (ödünçte mi?) alanı false olarak ayarlanır.**
3. **borrowDate alanı "-" olarak varsayılan değere atanır.**
4. **Girilen bilgiler dosyaya binary formatta yazılır.**
5. **Dosya kapatılır ve kullanıcıya bilgi verilir.**

void listBooks()

**Amaç: Dosyadaki tüm kitapları listelemek.**

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

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

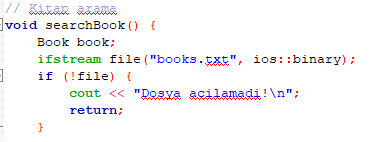
Yapay zeka tarafından oluşturulan içerik yanlış olabilir.

**Adımlar:**

1. **books.txt dosyası okunur.**
2. **Her kitap için ID, başlık, yazar, ödünç durumu ve varsa ödünç tarihi ekrana yazdırılır.**
3. **Ek olarak, kitabın ismi karakter karakter ([K][i][t][a][p]) şeklinde de gösterilir.**
4. **Hiç kitap yoksa, kullanıcı bilgilendirilir.**

**void searchBook()**

**Amaç: Kitapları ID’ye veya kitap adına göre aramak.**

****

1. **books.txt dosyası hem okumaya hem yazmaya açılır (fstream).**
2. **Kullanıcıdan ödünç alınacak kitap ID’si alınır.**
3. **Dosya taranır; eşleşen kitap bulunursa:**
4. **Zaten ödünçteyse kullanıcıya bildirilir.**
5. **Müsaitse: isBorrowed = true yapılır, kullanıcıdan tarih (GG/AA/YYYY) istenir.**
6. **Bu bilgiler güncellenip aynı konuma tekrar yazılır (seekp ile geri gidilir).**
7. **Kitap bulunamazsa kullanıcı bilgilendirilir.,**

**void borrowBook()**

**Amaç: Kullanıcının belirttiği kitabı ödünç almak.**

**metin, ekran görüntüsü, yazı tipi 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.**

**Adımlar:**

1. **books.txt dosyası hem okumaya hem yazmaya açılır (fstream).**
2. **Kullanıcıdan ödünç alınacak kitap ID’si alınır.**
3. **Dosya taranır; eşleşen kitap bulunursa:**
   * **Zaten ödünçteyse kullanıcıya bildirilir.**
   * **Müsaitse: isBorrowed = true yapılır, kullanıcıdan tarih (GG/AA/YYYY) istenir.**
   * **Bu bilgiler güncellenip aynı konuma tekrar yazılır (seekp ile geri gidilir).**
4. **Kitap bulunamazsa kullanıcı bilgilendirilir.**

**void returnBook()**

**Amaç: Ödünç alınan bir kitabı iade etmek.**

**metin, elektronik donanım, ekran görüntüsü, yazılım 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.**

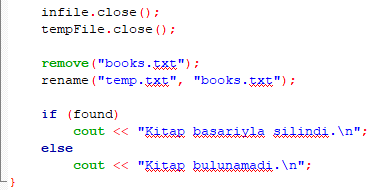
**Adımlar:**

1. **books.txt dosyası hem okumaya hem yazmaya açılır.**
2. **Kullanıcıdan iade etmek istediği kitap ID’si alınır.**
3. **Dosya taranır:**
   * **Eşleşen kitap bulunursa:**
     + **Eğer zaten kütüphanedeyse kullanıcı bilgilendirilir.**
     + **Ödünçteyse isBorrowed = false, borrowDate = "-" yapılır.**
     + **Bilgiler dosyada güncellenir.**
4. **Kitap bulunamazsa bilgi verilir.**

**void deleteBook()**

**Amaç: Belirtilen ID’ye sahip kitabı sistemden silmek.**

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

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.**

**Adımlar:**

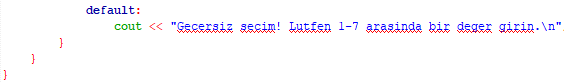
1. **Kullanıcıdan silinecek kitap ID’si alınır.**
2. **Tüm kitaplar okunur ve geçici bir temp.txt dosyasına yazılır.**
   * **Eğer ID eşleşiyorsa o kitap yazılmaz, yani silinir.**
3. **Asıl dosya silinir, geçici dosya books.txt olarak yeniden adlandırılır.**
4. **Kitap bulunduysa başarı mesajı, bulunamadıysa uyarı gösterilir.**

**void menu()**

**Amaç:** Kullanıcıya bir menü sunarak tüm sistem fonksiyonlarını kontrol etmek.

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

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.

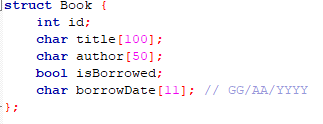
****

**İşleyiş:**

1. **Sürekli dönen bir while döngüsü vardır.**
2. **Kullanıcıdan 1-7 arasında bir seçim istenir:**
   * **1: Kitap ekle**
   * **2: Kitapları listele**
   * **3: Kitap ara**
   * **4: Kitap ödünç al**
   * **5: Kitap iade et**
   * **6: Kitap sil**
   * **7: Çıkış**
3. **Geçersiz seçimler uyarı verir, diğerleri ilgili fonksiyonu çağırır.**

**5. Diziler**

**Karakter Dizileri (Strings)**

****

**Dizi Kullanımı**

****

**6. İşaretçiler (Pointers)**

****

**reinterpret\_cast ile Book yapısı char\* tipine dönüştürülerek dosyaya yazılır.**

**7. Dosya İşlemleri**

**Dosyaya Yazma**

****

**Dosyadan Okuma**

****

**Dosya Güncelleme**

****

****

**Dosya Silme ve Yeniden Adlandırma**

****

**ALGORİTMA**

**GENEL ALGORİTMA SIRALAMASI**

**1. Program Başlangıcı (main)**

1.1 menu() fonksiyonu çağrılır.

**2. Ana Menü (menu() fonksiyonu)**

2.1 Sonsuz döngü başlatılır.  
2.2 Kullanıcıdan seçim alınır:

* 1 → addBook() çağrılır.
* 2 → listBooks() çağrılır.
* 3 → searchBook() çağrılır.
* 4 → borrowBook() çağrılır.
* 5 → returnBook() çağrılır.
* 6 → deleteBook() çağrılır.
* 7 → Program sonlandırılır.  
  2.3 Geçersiz seçimde uyarı verilir.

**FONKSİYONLARIN ALGORİTMALARI**

**3. Kitap Ekle (addBook)**

3.1 books.txt dosyası ekleme modunda açılır.  
3.2 Kitap bilgileri kullanıcıdan alınır:

* id, title, author  
  3.3 isBorrowed = false, borrowDate = "-" olarak ayarlanır.  
  3.4 Kitap dosyaya yazılır.  
  3.5 Başarı mesajı gösterilir.

**4. Kitapları Listele (listBooks)**

4.1 books.txt okuma modunda açılır.  
4.2 Dosya sonuna kadar tüm kitaplar tek tek okunur:

* id, title, author, isBorrowed, borrowDate yazdırılır.
* Kitap adı harf harf gösterilir.  
  4.3 Hiç kitap yoksa bilgilendirme yapılır.

**5. Kitap Ara (searchBook)**

5.1 books.txt okuma modunda açılır.  
5.2 Kullanıcıya arama türü sorulur:

* 1 → ID ile arama: ID eşleşirse kitap bilgisi yazdırılır.
* 2 → İsim ile arama: strcmp ile ad karşılaştırması yapılır.  
  5.3 Kitap bulunamazsa mesaj verilir.

**6. Kitap Ödünç Al (borrowBook)**

6.1 books.txt hem okuma hem yazma modunda açılır.  
6.2 Kullanıcıdan kitap ID'si alınır.  
6.3 Kitap bulunursa ve isBorrowed == false ise:

* isBorrowed = true yapılır.
* borrowDate kullanıcıdan alınır.
* seekp ile dosya konumu geri alınır ve güncellenir.  
  6.4 Zaten ödünçse uyarı verilir.  
  6.5 Kitap yoksa bilgilendirme yapılır.

**7. Kitap İade Et (returnBook)**

7.1 books.txt hem okuma hem yazma modunda açılır.  
7.2 Kullanıcıdan kitap ID'si alınır.  
7.3 Kitap bulunursa ve isBorrowed == true ise:

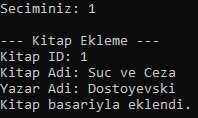
* isBorrowed = false yapılır.
* borrowDate = "-" yapılır.
* seekp ile dosya konumu geri alınır ve güncellenir.  
  7.4 Zaten kütüphanedeyse uyarı verilir.  
  7.5 Kitap yoksa mesaj gösterilir.

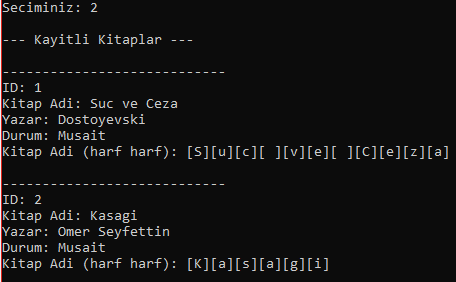
**8. Kitap Sil (deleteBook)**

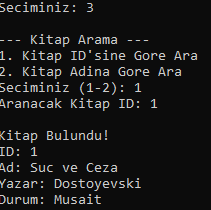
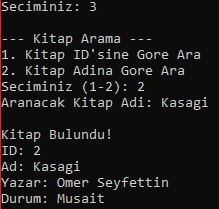
8.1 Kullanıcıdan silinecek kitap ID’si alınır.  
8.2 books.txt okuma modunda, temp.txt yazma modunda açılır.  
8.3 Dosya sonuna kadar kitaplar okunur:

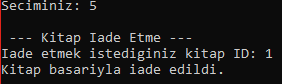
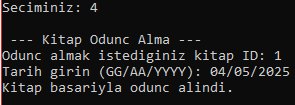
* Silinmek istenen kitap haricindekiler temp.txt'e yazılır.  
  8.4 books.txt silinir, temp.txt → books.txt olarak yeniden adlandırılır.  
  8.5 Başarı veya bulunamama mesajı verilir.

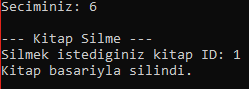
**EKRAN ÇIKTILARI**

****

****

**** ****

****

**** ****

**AKIŞ DİYAGRAMI**

