

**ALGORİTMA VE PROGRAMLAMAYA GİRİŞ DERSİ**

**2023-2024 GÜZ DÖNEMİ**

**SINIF/ŞUBE:** BMT-103 / 1.ŞUBE

**GRUP NO:** 01

**PROJE ADI:** KÜTÜPHANE

**ÖGRENCİLER/ÖĞRENCİ NO:**

**ÖMER FARUK AŞKIN 23181616758**

**MERT SEMİH SARIYERLİ 23181616001**

**ÖZGÜR TAŞKIRAN 23181616016**

**PROJE KONUSU:**

C kodu yardımıyla bir kütüphane oluşturuldu. Oluşturulan kütüphanede kitap ekleme kitap çıkarma eklenen kitapları görüntüleme, eklenen kitapları arama, kitap düzenleme, kitapları istendiği takdirde başka bir dosyaya yazdırma gibi seçenekler eklendi. Bu kod yardımıyla bilgisayarda kütüphane oluşturulabilir, kitaplar yazar adı, kitap adı, yayımlanma tarihi görünülecek şekilde gösterilebilir.

**NASIL YAPILDI?**

Öncelikle projedeki üç arkadaş toplanıp projeyi tartışıp bir akış şeması oluşturduk. Oluşturduktan sonra fonksiyonları belirleyip yazmaya başladık. Fonksiyonları yorum satırıyla birlikte yazdıktan farklı kütüphaneler yardımıyla farklı dosyalara yazdık. Kodda ortaya çıkan yanlışlıkları beraber tartışıp doğrusunu yazmaya çalıştık. Bunları yaptıktan sonra githubda projeyi oluşturup Clion’da yazdığımız kodları githuba yükledik. Bunları da yaptıktan sonra raporu yazdık. Rapordan sonra da akış şemasını oluşturduk. En son birkaç kez daha kodu gözden geçirdikten sonra projeyi teslim ettik.

**AKIŞ ŞEMASI:**

metin, diyagram, plan, teknik çizim içeren bir resim

Açıklama otomatik olarak oluşturuldu

**EKRAN ÇIKTISI:**

**|-------------------------------|**

**| Kutuphane Yonetim Sistemi |**

**|-------------------------------|**

**| 1. Kitap Ekleme |**

**| 2. Kitap Goruntuleme |**

**| 3. Kitap Arama |**

**| 4. Kitap Cikarma |**

**| 5. Kitap Duzenleme |**

**| 6. Kitaplari Dosyaya Yaz |**

**| 7. Cikis |**

**|-------------------------------|**

**Tercihinizi giriniz:1**

**Kitap basligini giriniz:Pc**

**Yazar giriniz:Muhendislik**

**Yayimlanma yilini giriniz(Rakamlarla):2024**

**Sayfa sayisini girin**

**iz(Rakamlarla):200**

**Kitap basariyla eklendi! ID: 808**

***C KODLARI:***

***#include <stdio.h>***

***#include <stdlib.h>***

***#include <string.h>***

***#include <time.h>***

***#define MAX\_kitaplar 100 //sabit global değişkenleri tanımlamak için kullandım***

***struct kitap{ //struct yapısı, kitap adında bir veri yapısı tanımlar ve içinde çeşitli kitap özelliklerini saklamak için kullandik.***

***int id;***

***char baslik[100];***

***char yazar[100];***

***int yil;***

***int sayfa\_sayisi;***

***};***

***struct kitap kutuphane[MAX\_kitaplar];***

***int kitapsayi = 0;***

***void eklekitap() {***

***if (kitapsayi < MAX\_kitaplar) {***

***struct kitap yenikitap;***

***srand(time(0)); //random id oluşturmak için srand yapısını kullandık.***

***yenikitap.id = rand() % 1000 + 1;***

***printf("Kitap basligini giriniz: ");***

***getchar(); //önceki satır başı karakterini kullanmamak için kullandık.***

***fgets(yenikitap.baslik, sizeof(yenikitap.baslik), stdin);***

***yenikitap.baslik[strcspn(yenikitap.baslik, "\n")] = '\0'; //Kullanıcı tarafından girilen kitap başlığını yenikitap.baslik dizisine alır. fgets(), standart giriş akışından (stdin), yani kullanıcı tarafından girilen verilerden bir satır okumak için kullanılır. Bu durumda yenikitap.baslik, kullanıcıdan alınan kitap başlığını saklamak için bir dizi olarak kullanılır.***

***printf("Yazar giriniz: ");***

***fgets(yenikitap.yazar, sizeof(yenikitap.yazar), stdin);***

***yenikitap.yazar[strcspn(yenikitap.yazar, "\n")] = '\0';// aynı şekilde kullancı tarafından girilen yazari yenikitap.yazar dizisine alır.***

***printf("Yayimlanma yilini giriniz(Rakamlarla): ");***

***scanf("%d", &yenikitap.yil);***

***printf("Sayfa sayisini giriniz(Rakamlarla): ");***

***scanf("%d", &yenikitap.sayfa\_sayisi);***

***kutuphane[kitapsayi++] = yenikitap; // kitap eklendiğinde yenikitap sayısını arttıran kod.***

***printf("Kitap basariyla eklendi! ID: %d\n", yenikitap.id);***

***} else {***

***printf("Kutuphane kapasitesi doldu baska kitap eklenemiyor.\n");***

***}***

***}***

***void kitaplariDosyayaYaz(const char \*kitaplar1) { // yazdığımız kitapları dosyaya eklemek için bir fonksiyon yazdık.***

***FILE \*dosya = fopen(kitaplar1, "w"); //burda kitaplar1 adinda bir yazma modunda dosya oluşturduk.***

***if (dosya == NULL) { //dosya açma işlemi NULL yani başarısız olursa dosya acilamadı yazdırdık.***

***printf("Dosya acilamadi.\n");***

***return;***

***}***

***for (int i = 0; i < kitapsayi; ++i) { // dizideki kitap sayısı kadar dönerek her bir kitabın bilgilerini dosyaya yazmayı sağlar.***

***fprintf(dosya, "%d;%s;%s;%d;%d\n", kutuphane[i].id, kutuphane[i].baslik, kutuphane[i].yazar, kutuphane[i].yil, kutuphane[i].sayfa\_sayisi);***

***}***

***fclose(dosya); //Dosya işlemi tamamlandıktan sonra, dosyayı kapatmak için fclose() fonksiyonu kullandık.***

***printf("Kitaplar %s dosyasina basariyla yazildi.\n", kitaplar1);***

***}***

***void kitapDuzenle() {***

***int duzenleID;***

***printf("Duzenlemek istediginiz kitabin ID'sini giriniz: ");***

***scanf("%d", &duzenleID);***

***int bulundu = 0;***

***for (int i = 0; i < kitapsayi; ++i) { //dizideki kitap sayısı kadar dönerek her bir kitabın ID'sini kontrol eder.***

***if (kutuphane[i].id == duzenleID) { //kutuphanede bulunan kitabın idsi ile bizim aradıgımız id uyuştugunda bu kod çalışır.***

***printf("Kitap bulundu! Lutfen yeni bilgileri giriniz:\n");***

***printf("Kitap basligini giriniz: ");***

***getchar(); //önceki satır başı karakterini kullanmamak için kullandık.***

***fgets(kutuphane[i].baslik, sizeof(kutuphane[i].baslik), stdin);***

***kutuphane[i].baslik[strcspn(kutuphane[i].baslik, "\n")] = '\0'; //Kullanıcı tarafından girilen yeni kitap başlığındaki olası satır sonu karakterini kaldırır ve yerine null karakter yani (\0) ekler. Bu, fgets() fonksiyonundan gelen satır sonu karakterini temizler ve veriyi düzenler.***

***printf("Yazar giriniz: ");***

***fgets(kutuphane[i].yazar, sizeof(kutuphane[i].yazar), stdin);***

***kutuphane[i].yazar[strcspn(kutuphane[i].yazar, "\n")] = '\0';//kullancı tarafından girilen yeni yazar ismini char ile kodlagımız için satır sonu karakterini kaldırır ve yerine null karakter ekler.***

***printf("Yayimlanma yilini giriniz: ");***

***scanf("%d", &kutuphane[i].yil);***

***printf("Sayfa sayisini giriniz: ");***

***scanf("%d", &kutuphane[i].sayfa\_sayisi);***

***printf("Kitap bilgileri guncellendi!\n");***

***bulundu = 1;***

***break;***

***}***

***}***

***if (!bulundu) {***

***printf("Kitap kutuphanede bulunamadi.\n");***

***}***

***}***

***void kitapgörünümü() {***

***if (kitapsayi == 0) {***

***printf("Kutuphanede kitap yok.\n");***

***} else {***

***printf("Kitap kutuphanede bulunuyor.:\n"); //kitap kutuphanede bulundugunda sırasiyla alt satırdakileri yazar.***

***printf("ID\tBaslik\tYazar\tYil\tSayfa Sayisi\n");***

***for (int i = 0; i < kitapsayi; ++i) {***

***printf("%d\t%s\t%s\t%d\t%d\n", kutuphane[i].id, kutuphane[i].baslik, kutuphane[i].yazar, kutuphane[i].yil, kutuphane[i].sayfa\_sayisi);***

***}***

***}***

***}***

***void arakitap() {***

***int araID;***

***printf("Kitap ID'sini giriniz.: ");***

***scanf("%d", &araID);***

***int bulundu = 0;***

***for (int i = 0; i < kitapsayi; ++i) {***

***if (kutuphane[i].id == araID) {***

***printf("kitap bulundu!\n");***

***printf("ID: %d\nBaslik: %s\nyazar: %s\nYil: %d\n", kutuphane[i].id, kutuphane[i].baslik, kutuphane[i].yazar, kutuphane[i].yil);***

***bulundu = 1;***

***break;***

***}***

***}***

***if (!bulundu) {***

***printf("Kitap kutuphanede bulunamadi.\n");***

***}***

***}***

***void cikarkitap() {***

***int cikarID;***

***printf("Cikarmak istediginiz kitabin ID'sini giriniz: ");***

***scanf("%d", &cikarID);***

***int bulundu = 0;***

***for (int i = 0; i < kitapsayi; ++i) { //dizideki kitap sayısı kadar dönerek her bir kitabın ID'sini kontrol eder.***

***if (kutuphane[i].id == cikarID) {***

***bulundu = 1; //eğer girilen id kütühanedeki id ile uyusursa bulundu 1e eşit olur ve çıkarılmak istenen kitabın indisinden başlayarak kütüphanedeki kitapları bir kaydırma işlemi yaparak bir önceki indise kopyalar. Bu, silinmek istenen kitabın listeden çıkarılmasını sağlar.***

***for (int j = i; j < kitapsayi - 1; ++j) {***

***kutuphane[j] = kutuphane[j + 1];***

***}***

***kitapsayi--;***

***printf("Kitap basariyla cikarildi!\n");***

***break;***

***}***

***}***

***if (!bulundu) {***

***printf("Kitap kutuphanede bulunamadi.\n");***

***}***

***}***

***int main() {***

***int tercih; //***

***do {***

***printf("\n|-------------------------------|\n");***

***printf("| Kutuphane Yonetim Sistemi |\n");***

***printf("|-------------------------------|\n");***

***printf("| 1. Kitap Ekleme |\n");***

***printf("| 2. Kitap Goruntuleme |\n");***

***printf("| 3. Kitap Arama |\n");***

***printf("| 4. Kitap Cikarma |\n");***

***printf("| 5. Kitap Duzenleme |\n");***

***printf("| 6. Kitaplari Dosyaya Yaz |\n");***

***printf("| 7. Cikis |\n");***

***printf("|-------------------------------|\n");***

***printf("Tercihinizi giriniz: ");***

***scanf("%d", &tercih);***

***switch (tercih) {***

***case 1:***

***eklekitap();***

***break;***

***case 2:***

***kitapgörünümü();***

***break;***

***case 3:***

***arakitap();***

***break;***

***case 4:***

***cikarkitap();***

***break;***

***case 5:***

***kitapDuzenle();***

***break;***

***case 6:***

***kitaplariDosyayaYaz("kitaplar.txt"); // Kitapları "kitaplar.txt" dosyasına yaz***

***break;***

***case 7:***

***printf("Programdan cikiliyor!\n");***

***break;***

***default:***

***printf("Hatali secim.Lutfen dogru rakami giriniz(1-7).\n");***

***}***

***} while (tercih != 7);***

***return 0;***

***}***