PROGRAMLAMAYA GİRİŞ DERSİ

PROJE ÖDEVİ

Projede bir kütüphanede kitap ödünç verme sisteminin takibi için c++ dosyalama ifadeleri kullanılarak gerçekleştirilecektir.

Okuyucu bilgileri(okuyucu.txt):

TC NO

ADI

SOYADI

üye No

TELEFON

DOĞUM TARİHİ

Görevi

Alabileceği kitap sayısı

Kitap bilgileri(Kitaplar.txt):

ISBN

Kitap İsmi

Yazar adı

Yazar soyadı

Konu

Tür

Sayfasayısı

Odunc verilen dosya (Odunc.txt)

ISBN

TCNO

Odunc tarihi

Donus tarihi

şeklindedir.

Program çalıştığında aşağıdaki menü seçenekleri kullanıcının karşısına çıkmalıdır.

1. Okuyucu kaydı
2. Okuyucu kaydı güncelleme
3. Okuyucu silme( Okuyucu silindiğinde okuyucunun ödünçleri geri dönmelidir.)
4. Okuyucu üzerindeki kitaplar listesi
5. Okuyucu kitap ödünç alma
6. Okuyucu kitap geri döndürme
7. Kitap ekleme
8. Kitap silme
9. Kitap düzeltme

## Çoklu dosya yönetimi

Uygulamada tek bir dosya kullanımı genellikle yetersiz kalmaktadır. Gerek veri tekrarını önlemek, veri organizasyonunu sağlamak üzere çok sayıda ve bir biri ile ilişkili dosyalarda verilerimizi tutarız. Örneğin hastaların kayıtlarının tutulmasında hastanın değişmeyen ad, soy ad, tel, doğum tarihi vb. bilgileri yanında, hastanın her muayene olduğundaki bilgileri vardır. Muayene bilgileri hasta için sık değişen bir veridir. Ayrı bir dosyada tutulması zorululuk gerektirir. Bu durumda en azından iki farklı dosya ile veri manipülasyonu gerçekleştirilmelidir. Şekil de dosyalar için muhtemel alanlar verilmiştir.

**Muayene Bilgileri**

Tc no

Tarih

Saat

Doktor adı

Teshis

İlaçlar

Tahliller

Vb

**Hasta Bilgileri**

Tc no

Hasta kayıt no

Adı

Soy adı,

Telefonu

Doğum tarihi,

Adres

Vb

(TC NO) her iki dosyadada yer almaktadır. Bu bir veri tekrarı gibi görünsede bize iki dosya arasındaki ilişkiyi kurmamızı sağlamada yardımcı olacaktır. Örneğin, Hastanın genel bilgilerini bir kere kayıt ettiğimizde, hasta her seferinde muayeneye geldiğinde sadece muayene bilgileri dosyasına muayene ile ilgili bilgiler kayıt eklenecektir. Kime ait olduğu da TC NO ile bilinecektir. Yani iki dosya arasında bilgilere ulaşımı TC no bilgisi sağlayacaktır.

//çoksayıda ilişkili dosya kullanımı

#include <windows.h>

#include <fstream>

#include <iostream>

#include <stdio.h>

#include <string>

using namespace std;

int main(int argc, char \*argv[])

{ // hasta.txt dosyasında yer alancak alanlar

int secim, i = 1;

string Hasta\_Adi;

string Hasta\_Soyadi;

string Tc;

string Tel;

int Hasta\_Yas;

char Hasta\_cinsiyet;

//muayene.txt dosyasında yer alacak alanlar

//string Tc; iki dosya arasındaki ilişkiyi sağlamak için

string tarih;

string DK\_Adi;

string Teshis;

double Ucret;

char cevap = 'e';

do

{

cout << " Ne yapmak istiyorsunuz ? " << endl;

cout << " 1) Hasta kayıt ekleme " << endl;

cout << " 2) Randevu kayıt " << endl;

cout << " 3) programdan çıkış " << endl;

cin >> secim;

} while (secim != 1 && secim != 2 && secim != 3);

if (secim == 1)

{

ofstream DosyaYaz;

DosyaYaz.open("Hasta.txt", ios::app);

do

{

cout << "\n Hasta TC :"; cin >> Tc;

cout << "\n Hasta adi :"; cin >> Hasta\_Adi;

cout << "\n Hasta soyadi :"; cin >> Hasta\_Soyadi;

cout << "\n Hasta Yasi :"; cin >> Hasta\_Yas;

cout << "\n telefonu :"; cin >> Tel;

cout << "\n cinsiyet :"; cin >> Hasta\_cinsiyet;

DosyaYaz << Tc << " " << Hasta\_Adi << ' ' << Hasta\_Soyadi << ' '

<< Hasta\_Yas << ' ' << Tel << ' '

<< Hasta\_cinsiyet << "\n";

cout << "\n baska kayit yapacak misin?(e/h) "; cin >> cevap;

} while (!(cevap == 'h'));

DosyaYaz.close();

cout << "dosyaya yazim tamamlandi. ";

}

if (secim == 2)

{

string TC;

cout << " Hasta TC :"; cin >> TC;

ifstream DosyaOku("Hasta.txt");

while (!DosyaOku.eof())

{

DosyaOku >> Tc >> Hasta\_Adi >> Hasta\_Soyadi

>> Hasta\_Yas >> Tel >> Hasta\_cinsiyet;

if (TC == Tc)

{

cout << "\n Hasta Bilgileri \n";

cout << " TC no :" << Tc << endl;

cout << "adi :" << Hasta\_Adi << endl;

cout << "soy adi :" << Hasta\_Soyadi << endl;

cout << "Yaşi :" << Hasta\_Yas << endl;

cout << "telefonu :" << Tel << endl;

cout << "cinsiyeti:" << Hasta\_cinsiyet << endl;

cout << " randevu istiyor mu? (e/h)" << endl;

cin >> cevap;

if (cevap == 'e')

{

DosyaOku.close();

ofstream DosyaYaz;

DosyaYaz.open("Muayene.txt", ios::app);

cout << "\n Randevu Bilgileri \n";

cout << "\n Tarih :"; cin >> tarih;

cout << "\n Doktor Adi :"; cin >> DK\_Adi;

cout << "\n Teshis :"; cin >> Teshis;

cout << "\n Muayene Ucreti :"; cin >> Ucret;

DosyaYaz << Tc << " " << tarih << ' ' << DK\_Adi << ' '

<< Teshis << ' ' << Ucret << "\n";

DosyaYaz.close();

system("PAUSE");

exit(1);

}

else

{

exit(1);

}

}

}

}

system("PAUSE");

return EXIT\_SUCCESS;

}

Hastanın hangi tarihlerde muayene olduğunu sorgulamak istediğimizde, önce TC no ile hasta “Hasta.txt” dosyasında sorgulanır.Hasta dosyada kayıtlı ise, ekrana getirilir.Hastanın muayene bilgileri “Muayene.txt” dosyasında yer aldığından dolayı, hastanın TC no ile muayene bilgileri “Muayene.txt” dosyasında araştırılır.Hasta farklı zamanlarda birden fazla muayene olmuş ise her birisi ekrana getirilecektir.Aşağıdaki kod bloğu böyle bir sorgulamayı gerçekleştirmek için kullanılabilir.

if (secim == 3)

{

string TC;

cout << " Hasta TC :"; cin >> TC;

ifstream DosyaOku("Hasta.txt");

while (!DosyaOku.eof())

{

DosyaOku >> Tc >> Hasta\_Adi >> Hasta\_Soyadi

>> Hasta\_Yas >> Tel >> Hasta\_cinsiyet;

if (TC == Tc)

{

cout << "\n Hasta Bilgileri \n";

cout << " TC no :" << Tc << endl;

cout << " adi :" << Hasta\_Adi << endl;

cout << " soy adi :" << Hasta\_Soyadi << endl;

cout << " Yaşi :" << Hasta\_Yas << endl;

cout << " telefonu :" << Tel << endl;

cout << " cinsiyeti:" << Hasta\_cinsiyet << endl;

TC = Tc;

DosyaOku.close();

ofstream DosyaYaz;

DosyaOku.open("Muayene.txt", ios::in);

cout << " randevu bilgileri \n" << endl;

cout << setw(11) << " TC " << setw(10) << " Tarih " << setw(12) << " Doktor Adi " << setw(10)

<< " Teshis " << setw(6) << " Ucret" << endl;

while (!DosyaOku.eof())

{

DosyaOku >> Tc >> tarih >> DK\_Adi >> Teshis >> Ucret;

if (TC == Tc)

{

cout << setw(11) << Tc << setw(10) << tarih << setw(12) << DK\_Adi << setw(10)

<< Teshis << setw(6) << Ucret << endl;

}

}

DosyaYaz.close();

system("PAUSE");

exit(1);

}

}

}

**İstenenler:**

1. **Proje için gerekli işlemleri yapan C++ kaynak kod dosyası**
2. **Projede yapılan işlemlerin basit bir şekilde anlatıldığı rapor dosyası**

# Ödevin son teslim tarihi :

# Yaz okulunun Son ders gününde ders saatinde sözlü olarak sunulacaktır. Sisteme yüklenmeyen projeler sunulmaya alınmayacaktır. Sunulmayan projeler 0 olarak değerlendirilecektir.

UYULMASI GEREKEN KOD DÜZENİ

Her C++ dosyasının başında aşağıdaki yorum bloğu bulunacaktır. Yorum bulunmayan her C++ dosyası için 10 puan kırılacaktır. (**pdf üzerinden kopyalayıp yapıştırmanız problem çıkartabilir**)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* SAKARYA ÜNİVERSİTESİ

\*\* BİLGİSAYAR VE BİLİŞİM BİLİMLERİ FAKÜLTESİ

\*\* BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

\*\* PROGRAMLAMAYA GİRİŞİ DERSİ

\*\*

\*\* ÖDEV NUMARASI…...:

\*\* ÖĞRENCİ ADI...............:

\*\* ÖĞRENCİ NUMARASI.:

\*\* DERS GRUBU…………:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

Değişken isimleri anlamlı olmalıdır. Örneğin tek sayı adedini tutacak bir değişken için

|  |  |  |  |
| --- | --- | --- | --- |
| YANLIŞ | yerine | DOĞRU | yazılmalıdır. |
| int a; | int tekAdedi; |

Her bir küme parantezinin altındaki kodları yazmadan önce **tab** tuşuna basılarak boşluk bırakılmalıdır.

|  |  |  |
| --- | --- | --- |
| YANLIŞ |  | DOĞRU |
| if(true)  {  int a =6;  cout<<"Merhaba";  if(false)  cout<<"Nasilsin";  } |  | if(true)  {  int a =6;  cout<<"Merhaba";  if(false)  cout<<"Nasilsin";  } |

Her döngü ve koşul işlemlerinden önce yapılan işlem hakkında yorum yazılmalıdır

|  |  |  |
| --- | --- | --- |
| YANLIŞ |  | DOĞRU |
| int t = 0;  for(int i=0;i<100;i++)  t+=i; |  | int toplam = 0;  //1 den 100 e kadar olan sayılar toplanıyor  for(int i=0;i<100;i++)  toplam+=i; |

ÖDEV TESLİM KURALLARI

# TESLİM EDİLECEK DOKÜMANLAR

* Ödev için bir rapor dosyası hazırlanacaktır.
* Dosya ‘pdf’ formatına çevrilip yollanması gerekmektedir.

# DERLEYİCİ VE TESLİM TARİHİ

Derleme için **Visual Studio 2017** C++ derleyicisi kullanılacaktır.

**Zamanında teslim edilmeyen ödevler değerlendirilmeyecektir.**

**Teslim saati sistemde belirtilen tarih ve saattir**

KOD PAYLAŞMAK YASAKTIR

Ödevler bireyseldir ve verilen bütün ödevler karşılaştırılacaktır. Birbirine çok benzeyen ödevler **kopya** muamelesi görecektir. Öğretim üyesi kopya durumunda ödevi değerlendirmez veya gerekli soruşturma ve ceza işlemlerini başlatabilir.