# 

T.C.

CUMHURİYET ÜNİVERSİTESİ

MÜHENDİSLİK FAKÜLTESİ

BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

# öğrenci otomasyonu

## tutku gündüzlü

## 2014123020

### *Özet:*

C++ diliyle yazılmış bu proje; öğrenci, öğretim elemanı, ders ve not girişi modüllerinden oluşan basit bir öğrenci otomasyonudur. Metin dosyaları kullanılarak oluşturulan bu proje; kayıt ekleme, kayıt güncelleme, arama, listeleme ve trankskript oluşturma işlemlerini gerekleştirmektedir.

***Anahtar Kelimeler:*** öğrenci, kayıt, arama, güncelleme, transkript

*Abstract*

*This project is written in C ++ language; student, teaching staff, not a course, but a simple student automation. Text files using created this project; adding records, updating records, searching, listing and creating trankscripts.*

***Keywords:*** student, record, search, update, transcript

## Giriş

Pek çok farklı dilde yazılabilen öğrenci otomasyonu bu projede veritabanı kullanılmadan metin dosyaları üzerinden gerçekleştirilmiştir.

Bu proje genel olarak metin dosyasından veri okuma ve metin dosyasına veri yazma mantığına dayanmaktadır.

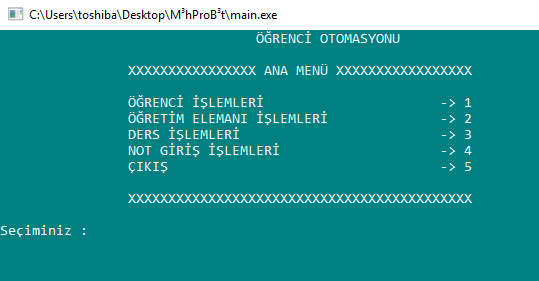
Kod üzerinde gerekli yerlerde açıklamalar bulunmaktadır ayrıca github üzerinden kodlara ulaşabilirsiniz.

Github linki KODLAR kısmındadır.

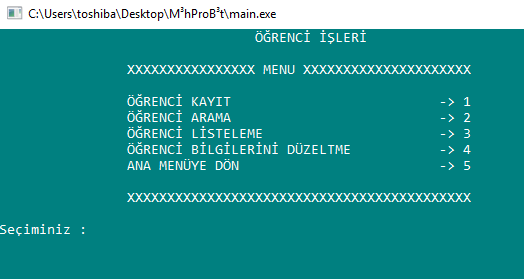
## UYGULAMA

Program C++ dili kullanılarak hazırlanmıştır.

Programda karşımıza ilk olarak öğrenci, öğretim elemanı, ders ve not girişi modüllerinin ve bir de çıkışın olduğu bir seçim ekranı çıkmaktadır. 1 ile 5 arasında yapılan seçime göre program ilgili modüle yönlendirmektedir. Aşağıda ana ekran görüntüsü bulunmaktadır :



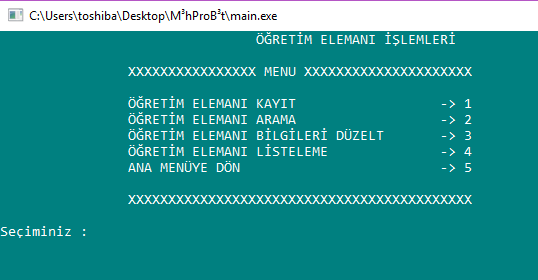
Seçimin “1” olması takdirde öğrenci işlemleri menüsü aşağıdaki gibidir :



Burada;

* “öğrenci kayıt” işlemi ile ogrenci.txt dosyasına kayıt eklenir,
* “öğrenci arama” işlemi ile öğrenci numarasına veya ismine göre arama işlemi gerçekleştirilir,
* “öğrenci listeleme” ile ögrenci.txt dosyasındaki veriler sınıfına veya bölümüne göre listeleme işlemi gerçekleştirilir,
* “öğrenci bilgilerini düzeltme” ile geçici bir dosya üzerine varolan veri değiştirilerek yazılır.

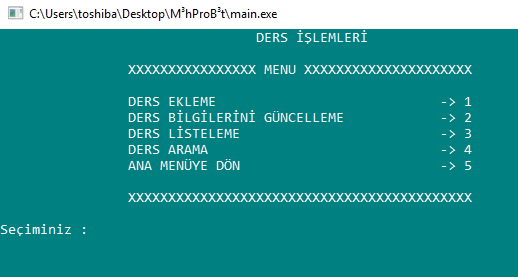
Seçimin “2” olması takdirde öğretim elemanı işlemleri menüsü aşağıdaki gibidir :



Burada;

* “öğretim elemanı kayıt” işlemi ile ogretimElemani.txt dosyasına kayıt eklenir,
* “öğretim elemanı arama” işlemi ile öğretim elemanı sicil numarasına göre arama işlemi gerçekleştirilir,
* “öğretim elemanı bilgileri düzelt” ile geçici bir dosya üzerine varolan veri değiştirilerek yazılır.
* “öğretim elemanı listeleme” ile ogretimElemani.txt dosyasındaki veriler bölüme göre listelenir.

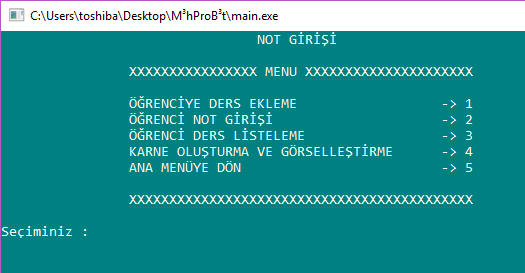
Seçimin “3” olması takdirde ders işlemleri menüsü aşağıdaki gibidir :



Burada;

* “ders ekleme” işlemi ile ders.txt dosyasına kayıt eklenir,
* “ders bilgilerini güncelleme” ile geçici bir dosya üzerine varolan veri değiştirilerek yazılır,
* “ders listeleme” işlemi ile dersler listelenir,
* “ders arama” ile ders adına göre arama işlemi gerçekleştirlir.

Seçimin “4” olması takdirde not giriş işlemleri menüsü aşağıdaki gibidir :



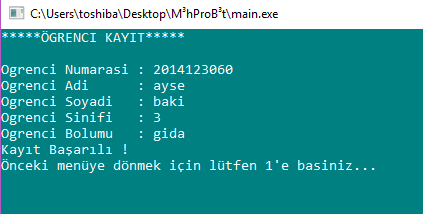
Burada ;

* “öğrenciye ders ekleme” işlemi ile dersEkle.txt ye kayıt eklenir.
* “öğrenci not girişi” işlemi ile öğrencinin almış olduğu derse not eklenir ve notEkle.txt ye yazılır.
* “öğrenci ders listeleme” işlemi ile dersEkle.txt okunarak ekranda listelenir.
* “karne oluşturma ve görselleştirme” işlemi ile ogrenci.txt ve notEkle.txt dosyalarında okunan veriler görsel düzen katılarak ekrana yazdırılır.

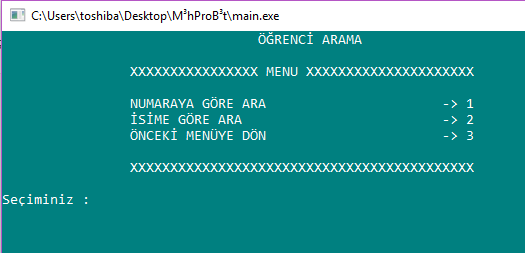
## SONUÇLAR

Proje kendisinden beklenen tüm beklenen işlemleri yerine getirmektedir.

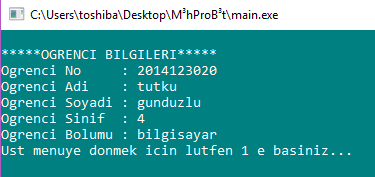
Aşağıda örnek olarak bir öğrenci kayıt işleminin sonucu gösterilmiştir :



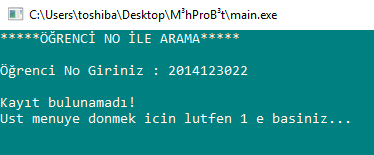
Aşağıda örnek olarak bir öğrenci arama işleminin sonucu gösterilmiştir :



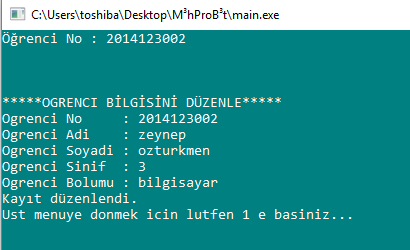
“numaraya göre ara” işlemi seçilirse program bir numara girmemizi bekler ve numara kayıtlı ise aşağıdaki gibi bir sonuç oluşur :



Eğer numara sistemde kayıtlı değil ise sonuç aşağıdaki gibi olur :

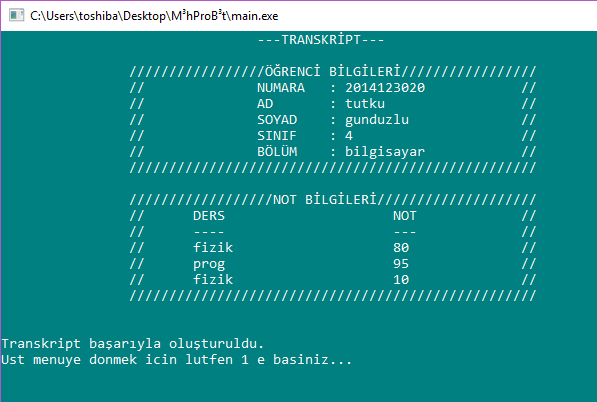


Yine öğrenci modülündeki kayıt güncelleme işlemi gerçekleştirilmek istendiğinde aşağıdaki sonuç elde edilir :



Diğer modüllerdeki işlemler de benzer şekildedir.

Son olarak not girişi menüsündeki “karne oluşturma ve görselleştirme” işlemi seçildiğinde ise aşağıda transkript ekrana yazılır :



## KAYnak kodlAR

Program kodlarının tamamı aşağıda C++ dilinde yazılmıştır.

GitHub linki : https://github.com/TutkuGunduzlu/Ogrenci-Otomasyonu/tree/master

#include <iostream>

#include <fstream> // dosya işlemleri için

#include <clocale> // türkçe karakter içn

#include <time.h> // sleep() fonksiyonu için

#include <unistd.h> // Türkçe karakter için

using namespace std;

int main(int argc, char\*\* argv) {

string no, isim, soyisim, sinif, bolum, noAra; // ogrenci.txt dosyasını okumak için değişkenler

string sicil, OEAd, OESoyad, OEBolum; // ogretimelemani.txt dosyasını okumak için değişkenler

string dersAd, dersKredi, dersBolum; // bolum.txt dosyasını okumak için değişkenler

string numaraDersEkle, dersEkleAd ; // dersEkle.txt dosyasını okumak için değişkenler

string noNotGiris, dersAdiNotGiris, puanNotGiris ; //notEkle.txt dosyasını okumak değişkenler

int donus, m=0;

enbas:

setlocale(LC\_ALL, "Turkish"); // Türkçe karakter için

system("COLOR 3f"); // ekran renklendirme için

int secim;

cout<<"\t\t\t\tÖĞRENCİ OTOMASYONU\n\n";

cout<<"\t\tXXXXXXXXXXXXXXXX ANA MENÜ XXXXXXXXXXXXXXXXX\n\n";

cout<<"\t\tÖĞRENCİ İŞLEMLERİ -> 1\n";

cout<<"\t\tÖĞRETİM ELEMANI İŞLEMLERİ -> 2\n";

cout<<"\t\tDERS İŞLEMLERİ -> 3\n";

cout<<"\t\tNOT GİRİŞ İŞLEMLERİ -> 4\n";

cout<<"\t\tÇIKIŞ -> 5\n";

cout<<"\n\t\tXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX";

cout<<"\n\nSeçiminiz : ";

cin>>secim;

system("cls");

switch(secim)

{

case 1: // öğrenci modülü

{

ogrenci:

cout<<"\t\t\t\tÖĞRENCİ İŞLERİ\n\n";

cout<<"\t\tXXXXXXXXXXXXXXXX MENU XXXXXXXXXXXXXXXXXXXXX\n\n";

cout<<"\t\tÖĞRENCİ KAYIT -> 1\n";

cout<<"\t\tÖĞRENCİ ARAMA -> 2\n";

cout<<"\t\tÖĞRENCİ LİSTELEME -> 3\n";

cout<<"\t\tÖĞRENCİ BİLGİLERİNİ DÜZELTME -> 4\n";

cout<<"\t\tANA MENÜYE DÖN -> 5\n"; cout<<"\n\t\tXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX";

cout<<"\n\nSeçiminiz : ";

cin>>secim;

system("cls");

switch(secim)

{

case 1: // öğrenci kayıt işlemi

{

cout<<"\*\*\*\*\*ÖGRENCI KAYIT\*\*\*\*\*"<<endl<<endl;

ofstream kaydet; // yazılacak dosyaya ait değişken tanımlanıyor

kaydet.open("ogrenci.txt",ios::out | ios::in | ios::app); // dosyaya yazma işlemi için mod belirleniyor

cout<<"Ogrenci Numarasi : "; cin>>no;

cout<<"Ogrenci Adi : "; cin>>isim;

cout<<"Ogrenci Soyadi : "; cin>>soyisim;

cout<<"Ogrenci Sinifi : "; cin>>sinif;

cout<<"Ogrenci Bolumu : "; cin>>bolum;

kaydet<<no<<" "<<isim<<" "<<soyisim<<" "<<sinif<<" "<<bolum<<endl; // kullanıcıdan alınan veriler dosyaya yazılıyor

kaydet.close(); // dosya kapama

cout<<"Kayıt Başarılı ! "<<endl<<"Önceki menüye dönmek için lütfen 1'e basiniz...";

cin>>donus;

if(donus==1){

system("cls"); // ekran sonraki işlem için temizleniyor

goto ogrenci; // kayıt işlemi bitince önceki menüye yönlendiriliyor

}

break;

}

case 2: // öğrenci arama işlemi

{

arama:

cout<<"\t\t\t\tÖĞRENCİ ARAMA\n\n";

cout<<"\t\tXXXXXXXXXXXXXXXX MENU XXXXXXXXXXXXXXXXXXXXX\n\n";

cout<<"\t\tNUMARAYA GÖRE ARA -> 1\n";

cout<<"\t\tİSİME GÖRE ARA -> 2\n";

cout<<"\t\tÖNCEKİ MENÜYE DÖN -> 3\n"; cout<<"\n\t\tXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX"; cout<<"\n\nSeçiminiz : ";

cin>>secim;

system("cls");

switch(secim)

{

case 1:// öğrenci numarasına göre arama

{

cout<<"\*\*\*\*\*ÖĞRENCİ NO İLE ARAMA\*\*\*\*\*"<<endl<<endl;

ifstream araNo; // dosya okuma için değişken belirleniyor

araNo.open("ogrenci.txt",ios::in); // dosya okuma modunda açılıyor

cout<<"Öğrenci No Giriniz : ";

cin>>noAra;

m=0;

do

{

if(araNo>>no>>isim>>soyisim>>sinif>>bolum) // veriler dosyadan okunuyor, veri varsa if çalışıyor

{

if (noAra == no) // girilen numara dosyada varsa if çalışıyor

{ system("cls");

cout<<endl<<"\*\*\*\*\*OGRENCI BILGILERI\*\*\*\*\*"<<endl;

cout<<"Ogrenci No : "<<no<<endl; cout<<"Ogrenci Adi : "<<isim<<endl;

cout<<"Ogrenci Soyadi : "<<soyisim<<endl; cout<<"Ogrenci Sinif : "<<sinif<<endl;

cout<<"Ogrenci Bolumu : "<<bolum<<endl;

m++;

}

}

} while (!araNo.eof()); // dosyanın sonuna kadar okuma yapılıyor

araNo.close(); //dosya kapatma

if(m==0) // m=0 ise hiç kayıt bulunamamıştır

cout<<endl<<"Kayıt bulunamadı!"<<endl;

cout<<"Ust menuye donmek icin lutfen 1 e basiniz...";

cin>>donus;

if(donus==1){

system("cls");

goto arama;

}

break;

}

case 2: // öğrenci ismine göre arama

{

cout<<"\*\*\*\*\*İSİM İLE ARAMA\*\*\*\*\*"<<endl<<endl;

ifstream araIsim;

araIsim.open("ogrenci.txt",ios::in);

string isimAra;

cout<<"Öğrencinin İsmini Giriniz : ";

cin>>isimAra;

m=0;

do

{

if(araIsim>>no>>isim>>soyisim>>sinif>>bolum)

{

if (isimAra == isim)

{

cout<<endl<<"\*\*\*\*\*OGRENCI BILGILERI\*\*\*\*\*"<<endl;

cout<<"Ogrenci No : "<<no<<endl;

cout<<"Ogrenci Adi : "<<isim<<endl;

cout<<"Ogrenci Soyadi : "<<soyisim<<endl;

cout<<"Ogrenci Sinif : "<<sinif<<endl;

cout<<"Ogrenci Bolumu : "<<bolum<<endl;

m++;

}

}

} while (!araIsim.eof());

araIsim.close();

if(m==0)

cout<<endl<<"Kayıt bulunamadı!"<<endl;

cout<<"Ust menuye donmek icin lutfen 1 e basiniz...";

cin>>donus;

if(donus==1){

system("cls");

goto arama;

}

break;

}

case 3: // önceki menü

{

goto ogrenci;

}

}

break;

}

case 3: // örenci listeleme işlemi

{

listele:

cout<<"\t\t\t\tÖĞRENCİ LİSTELEME\n\n";

cout<<"\t\tXXXXXXXXXXXXXXXX MENU XXXXXXXXXXXXXXXXXXXXX\n\n";

cout<<"\t\tBÖLÜME GÖRE LİSTELE -> 1\n";

cout<<"\t\tSINIFA GÖRE LİSTELE -> 2\n";

cout<<"\t\tÖNCEKİ MENÜYE DÖN -> 3\n";

cout<<"\n\t\tXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX";

cout<<"\n\nSeçiminiz : ";

cin>>secim;

system("cls");

switch(secim)

{

case 1: // öğrenci bölümüne göre

{

cout<<"\*\*\*\*\*BÖLÜME GÖRE ÖĞRENCİ LİSTELEME\*\*\*\*\*"<<endl<<endl;

ifstream listeleBolum;

listeleBolum.open("ogrenci.txt",ios::in);

string bolumListele;

cout<<"Bölüm : ";

cin>>bolumListele;

cout<<endl;

m=0;

do

{

if(listeleBolum>>no>>isim>>soyisim>>sinif>>bolum)

{

if (bolum == bolumListele)

{ cout<<endl<<endl<<"\*\*\*\*\*OGRENCI BILGILERI\*\*\*\*\*"<<endl;

cout<<"Ogrenci No : "<<no<<endl; cout<<"Ogrenci Adi : "<<isim<<endl;

cout<<"Ogrenci Soyadi : "<<soyisim<<endl;

cout<<"Ogrenci Sinif : "<<sinif<<endl; cout<<"Ogrenci Bolumu : "<<bolum<<endl;

m++;

}

}

} while (!listeleBolum.eof());

listeleBolum.close();

if(m==0)

cout<<endl<<"Bölüm bulunamadı!"<<endl;

cout<<"Ust menuye donmek icin lutfen 1 e basiniz...";

cin>>donus;

if(donus==1){

system("cls");

goto listele;

}

break;

}

case 2: // öğrenci sınıfına göre listeleme

{

cout<<"\*\*\*\*\*SINIFA GÖRE ÖĞRENCİ LİSTELEME\*\*\*\*\*"<<endl<<endl;

ifstream listeleSinif;

listeleSinif.open("ogrenci.txt",ios::in);

string sinifListele;

cout<<"Sınıf : ";

cin>>sinifListele;

cout<<endl;

m=0;

do

{

if(listeleSinif>>no>>isim>>soyisim>>sinif>>bolum)

{

if (sinif == sinifListele)

{ cout<<endl<<endl<<"\*\*\*\*\*ÖĞRENCİ BİLGİLERİ\*\*\*\*\*"<<endl;

cout<<"Ogrenci No : "<<no<<endl;

cout<<"Ogrenci Adi : "<<isim<<endl;

cout<<"Ogrenci Soyadi : "<<soyisim<<endl;

cout<<"Ogrenci Sinif : "<<sinif<<endl;

cout<<"Ogrenci Bolumu : "<<bolum<<endl;

m++;

}

}

} while (!listeleSinif.eof());

listeleSinif.close();

if(m==0)

cout<<endl<<"Sınıf bulunamadı!"<<endl;

cout<<"Ust menuye donmek icin lutfen 1 e basiniz...";

cin>>donus;

if(donus==1){

system("cls");

goto listele;

}

break;

}

case 3: // önceki menü

{

goto ogrenci;

}

}

break;

}

case 4: // öğrenci bilgilerini düzeltme işlemleri

{

// düzeltme işlemi için iki dosya gerekli: varolan ve geçiçi

ifstream okuEski;

ofstream yazYeni;

cout<<"Öğrenci No : ";

cin>>noAra;

cout<<endl;

okuEski.open("ogrenci.txt",ios::in); // varolan dosya okunur

yazYeni.open("geciciOgrenci.txt",ios::out | ios::in | ios::app); // geçici dosyaya yazılır

m=0;

do

{

if(okuEski>>no>>isim>>soyisim>>sinif>>bolum)

{

if (no == noAra)

{

cout<<endl<<endl<<"\*\*\*\*\*OGRENCI BİLGİSİNİ DÜZENLE\*\*\*\*\*"<<endl;

cout<<"Ogrenci No : "; cin>>no;

cout<<"Ogrenci Adi : "; cin>>isim;

cout<<"Ogrenci Soyadi : "; cin>>soyisim;

cout<<"Ogrenci Sinif : "; cin>>sinif;

cout<<"Ogrenci Bolumu : "; cin>>bolum;

// varolan dosyada bulunan bilgiler düzenlenince geçici dosyaya yazılır.

yazYeni<<no<<" "<<" "<<" "<<isim<<" "<<soyisim<<" "<<sinif<<" "<<bolum<<endl ;

m++;

}

else

{

yazYeni<<no<<" "<<" "<<" "<<isim<<" "<<soyisim<<" "<<sinif<<" "<<bolum<<endl ;

}

}

} while (!okuEski.eof());

okuEski.close();

yazYeni.close();

remove("ogrenci.txt"); // varolan dosya kaldırılır.

rename("geciciOgrenci.txt" , "ogrenci.txt"); // geçici dosyanın adı eski dosyayla değiştirilir

if(m==0)

cout<<endl<<"Düzenlenecek kayıt bulunamadı!"<<endl;

else

cout<<"Kayıt düzenlendi."<<endl;

cout<<"Ust menuye donmek icin lutfen 1 e basiniz...";

cin>>donus;

if(donus==1){

system("cls");

goto ogrenci;

}

break;

}

case 5:// önceki menü

{

goto enbas;

}

}

break;

}//case1

case 2: // öğretim elemanı modülü

{

ogretimElemani:

cout<<"\t\t\t\tÖĞRETİM ELEMANI İŞLEMLERİ\n\n";

cout<<"\t\tXXXXXXXXXXXXXXXX MENU XXXXXXXXXXXXXXXXXXXXX\n\n";

cout<<"\t\tÖĞRETİM ELEMANI KAYIT -> 1\n";

cout<<"\t\tÖĞRETİM ELEMANI ARAMA -> 2\n";

cout<<"\t\tÖĞRETİM ELEMANI BİLGİLERİ DÜZELT -> 3\n";

cout<<"\t\tÖĞRETİM ELEMANI LİSTELEME -> 4\n";

cout<<"\t\tANA MENÜYE DÖN -> 5\n";

cout<<"\n\t\tXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX";

cout<<"\n\nSeçiminiz : ";

cin>>secim;

system("cls");

switch(secim)

{

case 1: // öğretim elemanı kayıt

{

cout<<"\*\*\*\*\*ÖĞRETİM ELEMANI KAYIT\*\*\*\*\*"<<endl<<endl;

ofstream kaydetOE;

kaydetOE.open("ogretimElemani.txt",ios::out | ios::in | ios::app);

cout<<"Sicil No : "; cin>>sicil;

cout<<"Ad : "; cin>>OEAd;

cout<<"Soyad : "; cin>>OESoyad;

cout<<"Bölüm : "; cin>>OEBolum;

kaydetOE<<sicil<<" "<<OEAd<<" "<<OESoyad<<" "<<OEBolum<<endl;

kaydetOE.close();

cout<<"Kayıt Başarılı ! "<<endl<<"Önceki menüye dönmek için lütfen 1'e basiniz...";

cin>>donus;

if(donus==1){

system("cls");

goto ogretimElemani;

}

break;

}

case 2: // sicil no ile arama

{

cout<<"\*\*\*\*\*SİCİL NO İLE ARAMA\*\*\*\*\*"<<endl<<endl;

ifstream araSicil;

araSicil.open("ogretimElemani.txt",ios::in);

string sicilAra;

cout<<"Sicil No Giriniz : ";

cin>>sicilAra;

m=0;

do

{

if(araSicil>>sicil>>OEAd>>OESoyad>>OEBolum)

{

if (sicil == sicilAra)

{

system("cls");

cout<<endl<<"\*\*\*\*\*OGRENCI BILGILERI\*\*\*\*\*"<<endl;

cout<<"Sicil No : "<<sicil<<endl;

cout<<"Adı : "<<OEAd<<endl;

cout<<"Soyadı : "<<OESoyad<<endl;

cout<<"Bölüm : "<<OEBolum<<endl;

m++;

}

}

} while (!araSicil.eof());

araSicil.close();

if(m==0)

cout<<endl<<"Kayıt bulunamadı!"<<endl;

cout<<"Ust menuye donmek icin lutfen 1 e basiniz...";

cin>>donus;

if(donus==1){

system("cls");

goto ogretimElemani;

}

break;

}

case 3: // öğretim elemanı bilgilerini düzeltme

{

ifstream okuOE;

ofstream yazOE;

string sicilNo;

cout<<"Sicil No : ";

cin>>sicilNo;

cout<<endl;

okuOE.open("ogretimElemani.txt",ios::in);

yazOE.open("geciciOE.txt",ios::out | ios::in | ios::app);

m=0;

do

{

if(okuOE>>sicil>>OEAd>>OESoyad>>OEBolum)

{

if (sicil == sicilNo )

{

cout<<endl<<endl<<"\*\*\*\*\*OGRENTİM ELEMANI BİLGİLERİNİ DÜZENLEME\*\*\*\*\*"<<endl;

cout<<"Sicil No : "; cin>>sicil;

cout<<"Ad : "; cin>>OEAd;

cout<<"Soyad : "; cin>>OESoyad;

cout<<"Bölüm : "; cin>>OEBolum;

yazOE<<sicil<<" "<<OEAd<<" "<<OESoyad<<" "<<OEBolum<<endl ;

m++;

}

else

{

yazOE<<sicil<<" "<<OEAd<<" "<<OESoyad<<" "<<OEBolum<<endl ;

}

}

} while (!okuOE.eof());

okuOE.close();

yazOE.close();

remove("ogretimElemani.txt");

rename("geciciOE.txt" , "ogretimElemani.txt");

if(m==0)

cout<<endl<<"Düzenlenecek kayıt bulunamadı!"<<endl;

else

cout<<"Kayıt Düzenlendi."<<endl;

cout<<"Ust menuye donmek icin lutfen 1 e basiniz...";

cin>>donus;

if(donus==1){

system("cls");

goto ogretimElemani;

}

break;

}

case 4: // bölüme göre öğretim elemanı listeleme

{

cout<<"\*\*\*\*\*BÖLÜME GÖRE ÖĞRETİM ELEMANI LİSTELEME\*\*\*\*\*"<<endl<<endl;

ifstream listeleOE;

listeleOE.open("ogretimElemani.txt",ios::in);

string bolumBul;

cout<<"Bölüm : ";

cin>>bolumBul;

m=0;

do

{

if(listeleOE>>sicil>>OEAd>>OESoyad>>OEBolum)

{

if (OEBolum == bolumBul)

{

cout<<endl<<endl<<"\*\*\*\*\*ÖĞRETİM ELEMANI BİLGİLERİ LİSTELEME\*\*\*\*\*"<<endl;

cout<<"Sicil No : "<<sicil<<endl;

cout<<"Ad : "<<OEAd<<endl;

cout<<"Soyad : "<<OESoyad<<endl;

cout<<"Bölüm : "<<OEBolum<<endl;

m++;

}

}

} while (!listeleOE.eof());

listeleOE.close();

if(m==0)

cout<<endl<<"Bölüm bulunamadı!"<<endl;

cout<<"Ust menuye donmek icin lutfen 1 e basiniz...";

cin>>donus;

if(donus==1){

system("cls");

goto ogretimElemani;

}

break;

}

case 5:// önceki menü

{

goto enbas;

}

}

break;

}//case2

case 3: // ders modülü

{

ders:

cout<<"\t\t\t\tDERS İŞLEMLERİ\n\n";

cout<<"\t\tXXXXXXXXXXXXXXXX MENU XXXXXXXXXXXXXXXXXXXXX\n\n";

cout<<"\t\tDERS EKLEME -> 1\n";

cout<<"\t\tDERS BİLGİLERİNİ GÜNCELLEME -> 2\n";

cout<<"\t\tDERS LİSTELEME -> 3\n";

cout<<"\t\tDERS ARAMA -> 4\n";

cout<<"\t\tANA MENÜYE DÖN -> 5\n";

cout<<"\n\t\tXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX";

cout<<"\n\nSeçiminiz : ";

cin>>secim;

system("cls");

switch(secim)

{

case 1: // ders ekleme

{

cout<<"\*\*\*\*\*DERS KAYIT\*\*\*\*\*"<<endl<<endl;

ofstream dersEkle;

dersEkle.open("ders.txt",ios::out | ios::in | ios::app);

cout<<"Ders Adı : "; cin>>dersAd;

cout<<"Ders Kredisi : "; cin>>dersKredi;

cout<<"Dersin Bölümü : "; cin>>dersBolum;

dersEkle<<dersAd<<" "<<dersKredi<<" "<<dersBolum<<endl;

dersEkle.close();

cout<<"Kayıt Başarılı ! "<<endl<<"Önceki menüye dönmek için lütfen 1'e basiniz...";

cin>>donus;

if(donus==1){

system("cls");

goto ders;

}

break;

}

case 2: // der bilgilerini güncelleme

{

ifstream okuEskiDers;//dosya okuma ve dosyayı acma

ofstream yazYeniDers;

cout<<"Ders İsmi : ";

string dersIsim;

cin>>dersIsim;

cout<<endl;

okuEskiDers.open("ders.txt",ios::in);

yazYeniDers.open("geciciDers.txt",ios::out | ios::in | ios::app);

m=0;

do

{

if(okuEskiDers>>dersAd>>dersKredi>>dersBolum)

{

if (dersIsim == dersAd)

{

cout<<endl<<endl<<"\*\*\*\*\*DERS BİLGİSİNİ DÜZENLE\*\*\*\*\*"<<endl;

cout<<"Ders Adı : "; cin>>dersAd;

cout<<"Ders Kredisi : "; cin>>dersKredi;

cout<<"Dersin Bölümü : "; cin>>dersBolum;

yazYeniDers<<dersAd<<" "<<dersKredi<<" "<<dersBolum<<endl ;

m++;

}

else

{

yazYeniDers<<dersAd<<" "<<dersKredi<<" "<<dersBolum<<endl ;

}

}

} while (!okuEskiDers.eof());

okuEskiDers.close();

yazYeniDers.close();

remove("ders.txt");

rename("geciciDers.txt" , "ders.txt");

if(m==0)

cout<<endl<<"Düzenlenecek kayıt bulunamadı!"<<endl;

else

cout<<"Kayıt düzenlendi"<<endl;

cout<<"Ust menuye donmek icin lutfen 1 e basiniz...";

cin>>donus;

if(donus==1){

system("cls");

goto ders;

}

break;

}

case 3: // ders listeleme

{

cout<<"\*\*\*\*\*DERS LİSTELEME\*\*\*\*\*"<<endl<<endl;

ifstream listeleDers;

listeleDers.open("ders.txt",ios::in);

m=0;

do

{

if(listeleDers>>dersAd>>dersKredi>>dersBolum)

{

cout<<endl<<"\*\*\*\*\*DERS\*\*\*\*\*"<<endl;

cout<<"Ders Adı : "<<dersAd<<endl;

cout<<"Ders Kredisi : "<<dersKredi<<endl;

cout<<"Dersin Bölümü : "<<dersBolum<<endl;

m++;

}

} while (!listeleDers.eof());

listeleDers.close();

if(m==0)

cout<<endl<<"Ders bulunamadı!"<<endl;

cout<<endl<<"Ust menuye donmek icin lutfen 1 e basiniz...";

cin>>donus;

if(donus==1){

system("cls");

goto ders;

}

break;

}

case 4: // ders arama

{

cout<<"\*\*\*\*\*DERS ADI İLE ARAMA\*\*\*\*\*"<<endl<<endl;

ifstream araDers;

araDers.open("ders.txt",ios::in);

string dersAra;

cout<<"Ders Adını Giriniz : ";

cin>>dersAra;

m=0;

do

{

if(araDers>>dersAd>>dersKredi>>dersBolum)

{

if (dersAra == dersAd)

{

cout<<endl<<"\*\*\*\*\*DERS BİLGİLERİ\*\*\*\*\*"<<endl;

cout<<"Ders Adı : "<<dersAd<<endl;

cout<<"Ders Kredisi : "<<dersKredi<<endl;

cout<<"Dersin Bölümü : "<<dersBolum<<endl;

m++;

}

}

} while (!araDers.eof());

araDers.close();

if(m==0)

cout<<endl<<"Ders bulunamadı!";

cout<<endl<<"Ust menuye donmek icin lutfen 1 e basiniz...";

cin>>donus;

if(donus==1){

system("cls");

goto ders;

}

break;

}

case 5:// önceki menü

{

goto enbas;

}

}

break;

}//case 3

case 4: // not giriş modülü

{

notgiris:

cout<<"\t\t\t\tNOT GİRİŞİ\n\n";

cout<<"\t\tXXXXXXXXXXXXXXXX MENU XXXXXXXXXXXXXXXXXXXXX\n\n";

cout<<"\t\tÖĞRENCİYE DERS EKLEME -> 1\n";

cout<<"\t\tÖĞRENCİ NOT GİRİŞİ -> 2\n";

cout<<"\t\tÖĞRENCİ DERS LİSTELEME -> 3\n";

cout<<"\t\tKARNE OLUŞTURMA VE GÖRSELLEŞTİRME -> 4\n";

cout<<"\t\tANA MENÜYE DÖN -> 5\n";

cout<<"\n\t\tXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX";

cout<<"\n\nSeçiminiz : ";

cin>>secim;

system("cls");

switch(secim)

{

case 1: // öğrenciye ders ekleme

{

cout<<"\*\*\*\*\*ÖĞRENCİYE DERS EKLEME\*\*\*\*\*"<<endl<<endl;

ifstream okuOgrenci;

okuOgrenci.open("ogrenci.txt",ios::in);

ifstream okuDers;

okuDers.open("ders.txt",ios::in);

ofstream yazDersEkle;

yazDersEkle.open("dersEkle.txt",ios::in | ios::app);

string numara, derss;

int n=0;

cout<<"Öğrenci Numarasını Giriniz : ";

cin>>numara;

cout<<"Ders Adını Giriniz : ";

cin>>derss;

m=0;

do

{

if(okuOgrenci>>no>>isim>>soyisim>>sinif>>bolum)

{

if (numara == no) // girilen numara ogrenci.txt dosyasında varsa

{

do

{

if(okuDers>>dersAd>>dersKredi>>dersBolum )

{ if ( derss == dersAd) // girilen ders ders.txt dosyasında varsa

{

yazDersEkle<<numara<<" "<<derss<<endl; // dersEkle.txt dosyasına yazılır

n++;

}

}

} while (!okuDers.eof());

m++;

}

}

} while (!okuOgrenci.eof());

okuDers.close();

okuOgrenci.close();

yazDersEkle.close();

if(m==0) // öğrenci no sistemde kayıtlı değilse ekleme yapılamaz.

cout<<endl<<"Kayıtlı öğrenci bulunamadı!";

else if(n==0) // sistemdeki dersler haricinde ders girildiğinde ekleme yapılamaz.

cout<<endl<<"Yanlış(sistemde olmayan) ders adı girdiniz!";

else

cout<<endl<<"Öğrenciye ders eklendi";

cout<<endl<<"Ust menuye donmek icin lutfen 1 e basiniz...";

cin>>donus;

if(donus==1){

system("cls");

goto notgiris;

}

break;

}

case 2: //öğrencinin aldığı ders için not girişi

{

cout<<"\*\*\*\*\*ÖĞRENCİYE NOT EKLEME\*\*\*\*\*"<<endl<<endl;

ifstream okuOgrenci2;

okuOgrenci2.open("ogrenci.txt",ios::in);

ifstream okuDersEkle;

okuDersEkle.open("dersEkle.txt",ios::in);

ofstream yazNotEkle;

yazNotEkle.open("notEkle.txt",ios::in | ios::app);

string numara2, derss2;

int n=0, puan;

cout<<"Öğrenci Numarasını Giriniz : ";

cin>>numara2;

cout<<"Ders Adını Giriniz : ";

cin>>derss2;

cout<<"Öğrencini Notunu Giriniz : ";

cin>>puan;

m=0;

do

{

if(okuOgrenci2>>no>>isim>>soyisim>>sinif>>bolum)

{

if (numara2 == no) // girilen numara ogrenci.txt dosyasında varsa

{

do

{ if(okuDersEkle>>numaraDersEkle>>dersEkleAd ) {

if ( derss2 == dersEkleAd && numara2 == numaraDersEkle ) // öğrenci eğer belirtilen dersi almışsa ancak not eklenebilir { yazNotEkle<<numara2<<" "<<derss2<<" "<<puan<<endl; // öğrencinin notu notEkle.txt ye yazılır. n++;

break;

}

} } while (!okuDersEkle.eof());

m++;

}

}

} while (!okuOgrenci2.eof());

okuDersEkle.close();

okuOgrenci2.close();

yazNotEkle.close();

if(m==0) // girilen numara sistemde kayıtlı değilse not eklenemez.

cout<<endl<<"Kayıtlı öğrenci bulunamadı!";

else if(n==0) // girilen ders sistemde kayıtlı değilse not eklenemez.

cout<<endl<<"Yanlış(sistemde olmayan) ders adı girdiniz!";

else

cout<<endl<<"Öğrenciye not eklendi";

cout<<endl<<"Ust menuye donmek icin lutfen 1 e basiniz...";

cin>>donus;

if(donus==1){

system("cls");

goto notgiris;

}

break;

}

case 3: // öğrenci ders listeleme

{

cout<<"\*\*\*\*\*ÖĞRENCİ DERS LİSTELEME\*\*\*\*\*"<<endl<<endl;

ifstream ogrenciDersListele;

ogrenciDersListele.open("dersEkle.txt",ios::in);

string no3;

cout<<"Öğrenci No : ";

cin>>no3;

m=0;

do

{

if(ogrenciDersListele>>numaraDersEkle>>dersEkleAd)

{

if(no3 == numaraDersEkle) {

if(m==0){

cout<<endl<<"\*\*\*\*\*ALDIĞI DERSLER\*\*\*\*\*"<<endl;

} cout<<"Ders Adı : "<<dersEkleAd<<endl;

m++;

}

}

} while (!ogrenciDersListele.eof());

ogrenciDersListele.close();

if(m==0)

cout<<endl<<"Öğrenci bulunamadı!"<<endl;

cout<<endl<<"Ust menuye donmek icin lutfen 1 e basiniz...";

cin>>donus;

if(donus==1){

system("cls");

goto notgiris;

}

break; }

case 4: // öğrenci karnesi(traskript) oluşturma ve görselleşirme

{

cout<<"\*\*\*\*\*ÖĞRENCİ TRANSKRİPT OLUŞTURMA\*\*\*\*\*"<<endl<<endl;

ifstream notEkleTrans;

notEkleTrans.open("notEkle.txt",ios::in);

ifstream ogrenciTrans;

ogrenciTrans.open("ogrenci.txt",ios::in);

int n =0;

string no4;

cout<<"Öğrenci No:";

cin>>no4;

system("cls");

m=0;

// ilgili dosyalardan veriler okunur ve ekrana görselliği iyileştirilerek yazdırılır.

do

{

if(ogrenciTrans>>no>>isim>>soyisim>>sinif>>bolum)

{

if (no == no4) // girilen no sistemde kayıtlıysa transkript hazırlanır.

{ cout<<"\t\t\t\t---TRANSKRİPT---"<<endl<<endl; cout<<"\t\t/////////////////ÖĞRENCİ BİLGİLERİ/////////////////"<<endl; cout<<"\t\t//\t\tNUMARA : "<<no<<"\t\t //"<<endl; cout<<"\t\t//\t\tAD : "<<isim<<"\t\t //"<<endl; cout<<"\t\t//\t\tSOYAD : "<<soyisim<<"\t\t //"<<endl; cout<<"\t\t//\t\tSINIF : "<<sinif<<"\t\t //"<<endl; cout<<"\t\t//\t\tBÖLÜM : "<<bolum<<"\t\t //"<<endl; cout<<"\t\t///////////////////////////////////////////////////"<<endl<<endl; cout<<"\t\t//////////////////NOT BİLGİLERİ////////////////////"<<endl; cout<<"\t\t//\tDERS "<<"\t\t\t NOT\t\t //"<<endl; cout<<"\t\t//\t---- "<<"\t\t\t ---\t\t //"<<endl;

do { if(notEkleTrans>>noNotGiris>>dersAdiNotGiris>>puanNotGiris)

{

if ( noNotGiris == no4) { cout<<"\t\t//\t"<<dersAdiNotGiris<<"\t\t\t "<<puanNotGiris<<"\t\t //"<<endl;

n++; } }

}while (!notEkleTrans.eof());

m++;

}

}

} while (!ogrenciTrans.eof());

cout<<"\t\t///////////////////////////////////////////////////"<<endl<<endl;

ogrenciTrans.close();

notEkleTrans.close();

if(m==0 || n==0)

cout<<endl<<"Kayıtlı öğrenci bulunamadı!";

else

cout<<endl<<"Transkript başarıyla oluşturuldu.";

cout<<endl<<"Ust menuye donmek icin lutfen 1 e basiniz...";

cin>>donus;

if(donus==1){

system("cls");

goto notgiris;

}

break;

}

case 5:// önceki menü

{

goto enbas;

}

}

break;

}//case 4

case 5: // çıkış

{

return 0;

}// case 5

default:

{ // farklı bir rakam girlirse uyarı verilir

cout<<"Lutfen 1 le 5 arasinda bir rakam giriniz!"<<endl;

sleep(3); // 3 saniye beklenir

system("cls"); // ekran temizlenir

goto enbas; // ve programın en başına yönlendirlir

}

}

return 0;

}