

YILDIZ TEKNİK ÜNİVERSİTESİ  
ELEKTRİK-ELEKTRONİK FAKÜLTESİ  
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ



BLM-2031 YAPISAL PROGRAMLAMA

DÖNEM PROJESİ

Öğrenci Adı-Soyadı: ELİF GÜLERARSLAN

Öğrenci Numarası:21011007

Öğrenci Telefon Numarası:05396689988

[elif.gulerarslan@std.yildiz.edu.tr](mailto:elif.gulerarslan@std.yildiz.edu.tr)

Ders Yürütücüsü: Dr.Öğr.Üyesi Yunus Emre Selçuk

Ocak, 2023

## İÇİNDEKİLER

1)Struct ve Fonksiyon Tanımları

2)Main Fonksiyonu

a)Dosya Okuma

b)Switch-Case ile Menüde İşlem Seçimi

3) İşlemler

a)Yazar İşlemleri

b)Öğrenci İşlemleri

c)Kitap işlemleri

d)Kitap-Yazar İlişkilendirme

e)Öğrenci Kitap Ödünç Alma İşlemleri

4)Dosya Yazma İşlemleri

5)Kazanımlar

6)Kaynakça

## 1) Struct ve Fonksiyon Tanımları

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
```

Gerekli kütüphaneleri ekledikten sonra kullanılacak veri yapıları struct ile oluşturulur.

```
typedef struct Ogrenci {
    char ogrID[9];
    char ad[30];
    char soyad[30];
    int puan;
    struct Ogrenci *next;
    struct Ogrenci *prev;
}OGRENCI;

typedef struct Yazar {
    int yazarID;
    char yazarAd[30];
    char yazarSoyad[30];
    struct Yazar *next;
}YAZAR;

typedef struct KitapOrnek {
    char EtiketNo[20];
    char Durum[9];
    struct KitapOrnek *next;
}KITAPORNEK;

typedef struct Kitap {
    char kitapAdi[30];
    char ISBN[14];
    int adet;
    struct Kitap *next;
    struct KitapOrnek *head;
}KITAP;
```

```
typedef struct KitapYazar {
    char ISBN[14];
    int YazarID;
}KITAPYAZAR;

typedef struct Tarih {
    int gun:5;
    int ay:4;
    int yil:12;
}TARİH;

typedef struct KitapOdunc {
    char EtiketNo[20];
    char ogrID[9];
    int islemTipi:1;
    struct Tarih islemTarihi;
}KITAPODUNC;

typedef struct kityaz{
    struct KitapYazar* data;
    struct kityaz* next;
}KITYAZ;

typedef struct OduncAlma{
    struct KitapOdunc* p;
    struct OduncAlma* next;
}ODUNCALMA;
```

## Fonksiyon tanımları:

```
void ogrenciEkle(OGRENCI** head);
void ogrenciListele(OGRENCI** head);
void ogrenciGoruntule(int secim, OGRENCI** head);
void splitDataOgrenci(OGRENCI* head, char buff[255], char delimiter[2]);
OGRENCI* dosyaOkumaOgrenci(char* file_name);
void ogrenciSil(OGRENCI** head);
void dosyaYazdirOgrenci(OGRENCI* head, char* file_name);
void guncellemeIslemi(int secim, OGRENCI* tmp);
void ogrenciGuncelle(OGRENCI**head);
KITAP* dosyaOkumaKitap(char* file_name_kit);
void splitDataKitap(KITAP* head, char buff[255], char delimiter[2]);
int kitapKarsilastirma(char* buff, KITAP* head);
YAZAR* dosyaOkumaYazar(char* file_name);
void splitDataYazar(YAZAR* yaz, char buff[255], char delimiter[2]);
void yazarEkle(YAZAR** head);
void dosyaYazdirYazar(YAZAR* head, char* file_name);
void yazarSil(YAZAR** head);
void yazarGuncelle(YAZAR**head);
void guncellemeIslemiYazar(int secim, YAZAR* tmp);
void raftakiKitaplar(KITAP** head);
void dosyaYazdirKitap(KITAP* head, char* file_name);
KITAYAZ* dosyaOkumaKitapYazar(char* file_name_kit);
void kitapAdetleri(KITAP** head);
void splitDataKitapYazar(KITAPYAZAR* head, char buff[255], char delimiter[2]);
void dosyaYazdirKitapYazar(KITYAZ* head, char* file_name);
void kitapYazprint(KITYAZ* head, KITAP* head_kit, YAZAR* head_yaz);
void ISBN_kitap_bulma(KITAP* head, char* ISBN);
void yazar_bulma(YAZAR* head, int num);
ODUNCALMA* dosyaOkumaOdunc();
void splitDataOdunc(ODUNCALMA* head, char buff[255], char delimiter[2]);
void kitapOduncAlma(ODUNCALMA** h, KITAP** head);
void kitapOduncGuncel(KITAP**head, char etiket[20], char* str);
void yazarGoruntule(YAZAR** head);
```

## 2)Main Fonksiyonu

### a)Dosya Okuma

Verileri üzerinde işlem yapmak için kullanılacak olan linked listlerin herbirinin kendi türünde head pointerları oluşturulur ve ilgili fonksiyonlarla dosyalar okunarak linked listler oluşturulur.

```
int secim, kontrol=1;
OGRENCI* head_ogrenci;
KITAP* head_kitap;
YAZAR* head_yazar;
KITAYAZ* head_kitapYaz;
ODUNCALMA* head_odunc;
int secim_2; //öğrenci bilgisi görüntülemeye kullanılıyor

head_ogrenci=dosyaOkumaOgrenci("Ogrenciler.csv");
head_kitap=dosyaOkumaKitap("Kitaplar.csv");
kitapAdetleri(&head_kitap);
head_yazar=dosyaOkumaYazar("Yazarlar.csv");
head_kitapYaz=dosyaOkumaKitapYazar("KitapYazar.csv");
head_odunc=dosyaOkumaOdunc();
```

dosyaları okuyup linked listleri oluşturan fonksiyonlar :

```

OGRENCI* dosyaOkumaOgrenci(char* file_name){
    FILE* fp;
    fp=fopen(file_name, "r");
    char buff[255];

    OGRENCI* head =(OGRENCI*)malloc(sizeof(OGRENCI));

    if(head==NULL){
        return;
    }
    head->prev=NULL;
    OGRENCI* prev;
    OGRENCI* curr;
    if(fgets(buff, 255, fp)!=NULL){
        splitDataOgrenci(head, buff, ",");
    }else{
        return;
    }

}

ODUNCALMA* dosyaOkumaOdunc(){
    FILE* fp=fopen("KitapOdunc.csv","r");
    char buff[255];
    int kont;

    ODUNCALMA* head =(ODUNCALMA*)malloc(sizeof(ODUNCALMA));
    KITAPODUNC* k =(KITAPODUNC*)malloc(sizeof(KITAPODUNC));
    head->next=NULL;
    head->p=k;
    if(head==NULL || k==NULL){
        return;
    }

    if(fgets(buff, 255, fp)!=NULL){
        splitDataOdunc(head, buff, ",");
    }else{
        return;
    }
    ODUNCALMA* prev;
    prev=head;
    while(fgets(buff, 255, fp)!=NULL){
        ODUNCALMA* tmp =(ODUNCALMA*)malloc(sizeof(ODUNCALMA));
        KITAPODUNC* tmpk =(KITAPODUNC*)malloc(sizeof(KITAPODUNC));
        if(tmp==NULL || k==NULL){
            return;
        }

        tmp->p=tmpk;
        prev->next=tmp;
        splitDataOdunc(tmp, buff, ",");
        prev=tmp;
    }
    prev->next=NULL;
    fclose(fp);
    return head;
}

```

```

KITAP* dosyaOkumaKitap(char* file_name_kit){
    FILE* fp1;
    fp1=fopen(file_name_kit, "r");
    char buff[255];
    int kont;

    KITAP* head =(KITAP*)malloc(sizeof(KITAP));
    head->next=NULL;
    head->head=NULL;
    if(head==NULL){
        return;
    }

    if(fgets(buff, 255, fp1)!=NULL){
        splitDataKitap(head, buff, ",");
    }else{
        return;
    }
    KITAP* prev;
    prev=head;
    while(fgets(buff, 255, fp1)!=NULL){
        KITAP* tmp =(KITAP*)malloc(sizeof(KITAP));
        if(tmp==NULL){
            return;
        }

        tmp->head=NULL;
        prev->next=tmp;
        splitDataKitap(tmp,buff,",");
        prev=tmp;
    }
    prev->next=NULL;
    fclose(fp1);
    return head;
}

```



Aynı kitabın örneklerinin olduğu linked listi yapan fonksiyon:

```
void kitapAdetleri(KITAP** head){
    KITAP* tmp=(KITAP*)malloc(sizeof(KITAP));
    tmp=*head;
    int i;
    char new_ISBN[20], say[4];

    while(tmp!=NULL){
        KITAPORNEK* prev=(KITAPORNEK*)malloc(sizeof(KITAPORNEK));
        tmp->head=prev;
        strcpy(new_ISBN,tmp->ISBN);
        strcat(new_ISBN,"_");
        itoa(1, say, 10);
        strcat(new_ISBN, say);
        strcpy(prev->EtiketNo,new_ISBN);
        strcpy(prev->Durum,"rafta");

        for(i=2;i<=tmp->adet;i++){
            KITAPORNEK* yeni=(KITAPORNEK*)malloc(sizeof(KITAPORNEK));
            strcpy(new_ISBN,tmp->ISBN);
            strcat(new_ISBN,"_");
            itoa(i, say, 10);
            strcat(new_ISBN, say);
            strcpy(yeni->EtiketNo,new_ISBN);
            strcpy(yeni->Durum,"rafta");
            prev->next=yeni;
            prev=yeni;
        }
        prev->next=NULL;
        tmp=tmp->next;
    }
}
```



```

KITYAZ* dosyaOkumaKitapYazar(char* file_name_kit){
    FILE* fp1;
    fp1=fopen(file_name_kit, "r");
    char buff[255], buff2[255];
    int kont;

    KITYAZ* head =(KITYAZ*)malloc(sizeof(KITYAZ));
    KITAPYAZAR* p =(KITAPYAZAR*)malloc(sizeof(KITAPYAZAR));
    head->data= p;
    if(head==NULL || p==NULL){
        return;
    }

    if(fgets(buff, 255, fp1)!=NULL){
        splitDataKitapYazar(p, buff, ",");
    }else{
        return;
    }
    KITYAZ* prev;
    head->data=p;
    prev=head;

    while(fgets(buff, 255, fp1)!=NULL){
        KITYAZ* tmpb =(KITYAZ*)malloc(sizeof(KITYAZ));
        KITAPYAZAR* tmp =(KITAPYAZAR*)malloc(sizeof(KITAPYAZAR));
        if(tmp==NULL || tmpb==NULL){
            return;
        }

        prev->next=tmpb;
        splitDataKitapYazar(tmp,buff,",");
        tmpb->data=tmp;
        prev=tmpb;
    }
    prev->next=NULL;
    fclose(fp1);
    return head;
}

```

```

YAZAR* dosyaOkumaYazar(char* file_name){
    FILE* fp;
    fp=fopen(file_name, "r");
    char buff[255];

    YAZAR* head =(YAZAR*)malloc(sizeof(YAZAR));

    if(head==NULL){
        return;
    }
    YAZAR* prev;
    YAZAR* curr;
    if(fgets(buff, 255, fp)!=NULL){
        splitDataYazar(head, buff, ",");
    }else{
        return;
    }

    prev=head;
    while(fgets(buff, 255, fp)!=NULL){
        curr=(YAZAR*)malloc(sizeof(YAZAR));
        prev->next=curr;

        splitDataYazar(curr,buff,",");
        prev=curr;
    }
    prev->next=NULL;
    fclose(fp);
    return head;
}

```

Dosyayı okurken verileri parçalayan fonksiyonlar:

```

void splitDataOgrenci(OGRENCI* head, char buff[255], char delimiter[2]){
    char* token;

    token=strtok(buff, delimiter);
    strcpy(head->ogrID, token);

    token=strtok(NULL, delimiter);
    strcpy(head->ad,token);

    token=strtok(NULL, delimiter);
    strcpy(head->soyad,token);

    token=strtok(NULL, delimiter);
    head->puan=atoi(token);
}

```

```

void splitDataOdunc(ODUNCALMA* head, char buff[255], char delimiter[2]){
    char* token;

    token=strtok(buff, delimiter);
    strcpy(head->p->EtiketNo, token);

    token=strtok(NULL, delimiter);
    strcpy(head->p->ogrID,token);

    token=strtok(NULL, delimiter);
    head->p->islemTipi=atoi(token);

    token=strtok(NULL, ".");
    head->p->islemTarihi.gun=atoi(token);

    token=strtok(NULL, ".");
    head->p->islemTarihi.ay=atoi(token);
    token=strtok(NULL, "\n");
    head->p->islemTarihi.yil=atoi(token);
}

```

```

void splitDataKitap(KITAP* head, char buff[255], char delimiter[2]){
    char* token;

    token=strtok(buff, delimiter);
    strcpy(head->kitapAdi, token);

    token=strtok(NULL, delimiter);
    strcpy(head->ISBN,token);

    token=strtok(NULL, delimiter);
    head->adet=atoi(token);
}

```

```

void splitDataKitapYazar(KITAPYAZAR* head, char buff[255], char delimiter[2]){
    char* token;

    token=strtok(buff, delimiter);
    strcpy(head->ISBN, token);

    token=strtok(NULL, delimiter);
    head->YazarID=atoi(token);
}

```

```

void splitDataYazar(YAZAR* yaz, char buff[255], char delimiter[2]){
    char* token;

    token=strtok(buff, delimiter);
    yaz->yazarID=atoi(token);

    token=strtok(NULL, delimiter);
    strcpy(yaz->yazarAd,token);

    token=strtok(NULL, "\n");
    strcpy(yaz->yazarSoyad,token);
}

```

## b)Switch-Case ile Menüde İşlem Seçimi

Menü:

```

Ogrenci islemleri:
    1-Ogrenci ekle
    2-Ogrenci sil
    3-Ogrenci guncelle
    4-Ogrenci bilgisi görüntüle
    5-Kitap teslim etmememis ogrencileri listele
    6-Cezali ogrencileri listele
    7-Tum ogrencileri listele
    8-Kitap odunc al
    9-Kitap teslim et
Kitap islemleri:
    10-Kitap ekle
    11-Kitap sil
    12-Kitap guncelle
    13-Kitap bilgisi görüntüle
    14-Raftaki kitaplari listele
    15-Zamaninda teslim edilmeyen kitaplari listele
    16-Kitap-Yazar eslestir
    17-Kitabin yazarini guncelle
Yazar islemleri:
    18-Yazar ekle
    19-Yazar sil
    20-Yazar guncelle
    21-Yazar bilgisi görüntuleme

    22-programdan cikis

yapmak istediginiz islemin numarasini giriniz:

```

### 3)İşlemler

#### a)Yazar İşlemleri

##### i)Yazar Ekle

Konsoldan eklenişi:

```
Yazar islemleri:
 18-Yazar ekle
 19-Yazar sil
 20-Yazar guncelle
 21-Yazar bilgisi goruntuleme

 22-programdan cikis

yapmak istediginiz islemin numarasini giriniz:
18
sirasiyla ad, soyad giriniz:
elif
gulerarslan
```

Csv Dosyasına yazılışı:

|                       |  |
|-----------------------|--|
| 1,Michael,Jordan      |  |
| 2,Stephen,Boyd        |  |
| 3,Kalyanmoy,Deb       |  |
| 4,David,Johnson       |  |
| 5,Scott,Kirkpatrick   |  |
| 6,Lieven,Vandenberghe |  |
| 7,Fabian,Pedregosa    |  |
| 8,Jorge,Nocedal       |  |
| 9,elif,gulerarslan    |  |
|                       |  |

## ii)Yazar Sil

Konsolda ve csv dosyasındaki input ve çıktılar:

```
17-Kitabin yazarini guncelle
Yazar islemleri:
18-Yazar ekle
19-Yazar sil
20-Yazar guncelle
21-Yazar bilgisi görüntüleme

22-programdan cikis

yapmak istediginiz islemin numarasini giriniz:
19
silmek istediginiz yazarin adi:
elif
silmek istediginiz yazarin soyadi:
gulerarslan
Yazar silinmistir
```

|                       |  |  |
|-----------------------|--|--|
| 1,Michael,Jordan      |  |  |
| 2,Stephen,Boyd        |  |  |
| 3,Kalyanmoy,Deb       |  |  |
| 4,David,Johnson       |  |  |
| 5,Scott,Kirkpatrick   |  |  |
| 6,Lieven,Vandenberghe |  |  |
| 7,Fabian,Pedregosa    |  |  |
| 8,Jorge,Nocedal       |  |  |
|                       |  |  |
|                       |  |  |

Yazar ekle işleminde eklenen “elif gulerarslan” kişisi silinmiştir.

## iii)Yazar Güncelle

Konsolda ve csv dosyasındaki ad güncelleme için input ve çıktılar:

```
22-programdan cikis

yapmak istediginiz islemin numarasini giriniz:
20
bilgilerini guncellemek istediginiz yazarin adi:
David
bilgilerini guncellemek istediginiz yazarin soyadi:
Johnson
1-ad guncelleme
2-soyad guncelleme
3-guncelleme yapmak istemiyorum
yapmak istediginiz islemi seciniz:
1
yeni adi giriniz:Danny
```

|                       |  |  |
|-----------------------|--|--|
| 1,Michael,Jordan      |  |  |
| 2,Stephen,Boyd        |  |  |
| 3,Kalyanmoy,Deb       |  |  |
| 4,Danny,Johnson       |  |  |
| 5,Scott,Kirkpatrick   |  |  |
| 6,Lieven,Vandenberghe |  |  |
| 7,Fabian,Pedregosa    |  |  |
| 8,Jorge,Nocedal       |  |  |
|                       |  |  |
|                       |  |  |

Konsolda ve csv dosyasındaki soyad güncelleme için input ve çıktılar:

```
yapmak istediginiz islemin numarasini giriniz:
20
bilgilerini guncellemek istediginiz yazarin adi:
Danny
bilgilerini guncellemek istediginiz yazarin soyadi:
Johnson
1-ad guncelleme
2-soyad guncelleme
3-guncelleme yapmak istemiyorum
yapmak istediginiz islemi seciniz:
2
yeni soyadi giriniz:Leith
```

|                       |  |
|-----------------------|--|
| 1,Michael,Jordan      |  |
| 2,Stephen,Boyd        |  |
| 3,Kalyanmoy,Deb       |  |
| 4,Danny,Leith         |  |
| 5,Scott,Kirkpatrick   |  |
| 6,Lieven,Vandenberghe |  |
| 7,Fabian,Pedregosa    |  |
| 8,Jorge,Nocedal       |  |
|                       |  |
|                       |  |

#### iv)Yazar Bilgisi Görüntüle

Konsolda yazar görüntüleme için input ve çıktılar:

```
yapmak istediginiz islemin numarasini giriniz:
21
yazarin adini giriniz: Michael
yazarin soyadini giriniz: Jordan
yazarin:
    adi: Michael
    soyadi: Jordan
    ID: 1
```

Yazar görüntüleme seçildiğinde yazarın adı, soyadı ve ID numarası ekrana çıktı olarak yazılır.



## b)Öğrenci İşlemleri

Sadece tamamlanmış olan işlemlerin çıktıları yer alıyor bu bölümde.

### i)Öğrenci Ekleme

Konsolda ve csv dosyasındaki input ve çıktılar:

```
22 programdan çıkış
yapmak istediginiz islemin numarasini giriniz:
1
sirasiyla ad, soyad, ID giriniz:
elif
gulerarslan
12345678
```

|                                 |  |
|---------------------------------|--|
| 18011001,Geoffrey,Hinton,100    |  |
| 18011005,Yoshua,Bengio,100      |  |
| 18011010,Robert,Tibshirani,100  |  |
| 18011015,Kaiming,He,100         |  |
| 18011020,Ilya,Sutskever,100     |  |
| 17011001,Andrew,Zisserman,100   |  |
| 17011005,Francisco,Matorras,100 |  |
| 17011010,Ross,Girshick,100      |  |
| 17011015,Trevor,Hastie,100      |  |
| 17011020,Yann,LeCun,100         |  |
| 12345678,elif,gulerarslan,100   |  |

### ii)Öğrenci Silme

Konsolda ve csv dosyasındaki input ve çıktılar:

```
yapmak istediginiz islemin numarasini giriniz:
2
silme istediginiz ogrencinin adi:
Francisco
silme istediginiz ogrencinin soyadi:
Matorras
Ogrenci silinmistir
```

|                                |  |
|--------------------------------|--|
| 18011001,Geoffrey,Hinton,100   |  |
| 18011005,Yoshua,Bengio,100     |  |
| 18011010,Robert,Tibshirani,100 |  |
| 18011015,Kaiming,He,100        |  |
| 18011020,Ilya,Sutskever,100    |  |
| 17011001,Andrew,Zisserman,100  |  |
| 17011010,Ross,Girshick,100     |  |
| 17011015,Trevor,Hastie,100     |  |
| 17011020,Yann,LeCun,100        |  |
| 12345678,elif,gulerarslan,100  |  |

“Francisco Matorras” adlı öğrenci dosyadan ve linked listten silinmiştir.

### iii)Öğrenci Güncelle

Konsolda ve csv dosyasındaki ad güncelleme için input ve çıktılar:

```
yapmak istediginiz islemin numarasini giriniz:
3
bilgilerini guncellemek istediginiz ogrencinin adi:
Trevor
bilgilerini guncellemek istediginiz ogrencinin soyadi:
Hastie
1-ad guncelleme
2-soyad guncelleme
3-ID guncelleme
4-puan guncelleme
5-guncelleme yapmak istemiyorum
yapmak istediginiz islemi seciniz:
1
yeni adi giriniz:Travis
```

|          |                       |  |
|----------|-----------------------|--|
| 18011001 | Geoffrey,Hinton,100   |  |
| 18011005 | Yoshua,Bengio,100     |  |
| 18011010 | Robert,Tibshirani,100 |  |
| 18011015 | Kaiming,He,100        |  |
| 18011020 | Ilya,Sutskever,100    |  |
| 17011001 | Andrew,Zisserman,100  |  |
| 17011010 | Ross,Girshick,100     |  |
| 17011015 | Travis,Hastie,100     |  |
| 17011020 | Yann,LeCun,100        |  |
| 12345678 | elif,gulerarslan,100  |  |

Konsolda ve csv dosyasındaki soyad güncelleme için input ve çıktılar:

```
yapmak istediginiz islemin numarasini giriniz:
3
bilgilerini guncellemek istediginiz ogrencinin adi:
Travis
bilgilerini guncellemek istediginiz ogrencinin soyadi:
Hastie
1-ad guncelleme
2-soyad guncelleme
3-ID guncelleme
4-puan guncelleme
5-guncelleme yapmak istemiyorum
yapmak istediginiz islemi seciniz:
2
yeni soyadi giriniz:Hugo
```

|          |                       |  |
|----------|-----------------------|--|
| 18011001 | Geoffrey,Hinton,100   |  |
| 18011005 | Yoshua,Bengio,100     |  |
| 18011010 | Robert,Tibshirani,100 |  |
| 18011015 | Kaiming,He,100        |  |
| 18011020 | Ilya,Sutskever,100    |  |
| 17011001 | Andrew,Zisserman,100  |  |
| 17011010 | Ross,Girshick,100     |  |
| 17011015 | Travis,Hugo,100       |  |
| 17011020 | Yann,LeCun,100        |  |
| 12345678 | elif,gulerarslan,100  |  |

Konsolda ve csv dosyasındaki ID güncelleme için input ve çıktılar:

```
yapmak istediginiz islemin numarasini giriniz:
3
bilgilerini guncellemek istediginiz ogrencinin adi:
Travis
bilgilerini guncellemek istediginiz ogrencinin soyadi:
Hugo
1-ad guncelleme
2-soyad guncelleme
3-ID guncelleme
4-puan guncelleme
5-guncelleme yapmak istemiyorum
yapmak istediginiz islemi seciniz:
3
yeni ID giriniz:44444444
```

|          |                   |     |  |
|----------|-------------------|-----|--|
| 18011001 | Geoffrey,Hinton   | 100 |  |
| 18011005 | Yoshua,Bengio     | 100 |  |
| 18011010 | Robert,Tibshirani | 100 |  |
| 18011015 | Kaiming,He        | 100 |  |
| 18011020 | Ilya,Sutskever    | 100 |  |
| 17011001 | Andrew,Zisserman  | 100 |  |
| 17011010 | Ross,Girshick     | 100 |  |
| 44444444 | Travis,Hugo       | 100 |  |
| 17011020 | Yann,LeCun        | 100 |  |
| 12345678 | elif,gulerarslan  | 100 |  |
|          |                   |     |  |

Konsolda ve csv dosyasındaki puan güncelleme için input ve çıktılar:

```
yapmak istediginiz islemin numarasini giriniz:
3
bilgilerini guncellemek istediginiz ogrencinin adi:
Travis
bilgilerini guncellemek istediginiz ogrencinin soyadi:
Hugo
1-ad guncelleme
2-soyad guncelleme
3-ID guncelleme
4-puan guncelleme
5-guncelleme yapmak istemiyorum
yapmak istediginiz islemi seciniz:
4
yeni puani giriniz:90
```

|          |                   |     |  |
|----------|-------------------|-----|--|
| 18011001 | Geoffrey,Hinton   | 100 |  |
| 18011005 | Yoshua,Bengio     | 100 |  |
| 18011010 | Robert,Tibshirani | 100 |  |
| 18011015 | Kaiming,He        | 100 |  |
| 18011020 | Ilya,Sutskever    | 100 |  |
| 17011001 | Andrew,Zisserman  | 100 |  |
| 17011010 | Ross,Girshick     | 100 |  |
| 44444444 | Travis,Hugo       | 90  |  |
| 17011020 | Yann,LeCun        | 100 |  |
| 12345678 | elif,gulerarslan  | 100 |  |
|          |                   |     |  |

#### iv)Öğrenci bilgisi görüntüle

Konsolda öğrenci bilgisini ad-soyad ile görüntüleme için input ve çıktılar:

```
yapmak istediginiz islemin numarasini giriniz:
4
Ogrenci bilgisini ad-soyad ile gorunulemek icin: 1
Ogrenci bilgisini ID numarasi ile gorunulemek icin: 2
1
ogrencinin adini giriniz: Travis
ogrencinin soyadini giriniz: Hugo
ogrencinin:
    adi: Travis
    soyadi: Hugo
    numarasi: 44444444
    puani: 90
```

Konsolda öğrenci bilgisini ID ile görüntüleme için input ve çıktılar:

```
yapmak istediginiz islemin numarasini giriniz:
4
Ogrenci bilgisini ad-soyad ile gorunulemek icin: 1
Ogrenci bilgisini ID numarasi ile gorunulemek icin: 2
2
ogrencinin ID'sini giriniz: 17011001
ogrencinin:
    adi: Andrew
    soyadi: Zisserman
    numarasi: 17011001
    puani: 100
```

#### iv)Tüm Öğrencileri Listele

Konsolda ve csv dosyasındaki tüm öğrencileri listelemek için input ve çıktılar:

```
yapmak istediginiz islemin numarasini giriniz:
7
1. ogrenci:
    ad: Geoffrey
    soyad: Hinton
    numara: 18011001
    puan: 100
2. ogrenci:
    ad: Yoshua
    soyad: Bengio
    numara: 18011005
    puan: 100
3. ogrenci:
    ad: Robert
    soyad: Tibshirani
    numara: 18011010
    puan: 100
```

```
4. ogrenci:
    ad: Kaiming
    soyad: He
    numara: 18011015
    puan: 100
5. ogrenci:
    ad: Ilya
    soyad: Sutskever
    numara: 18011020
    puan: 100
6. ogrenci:
    ad: Andrew
    soyad: Zisserman
    numara: 17011001
    puan: 100
7. ogrenci:
    ad: Ross
    soyad: Girshick
    numara: 17011010
    puan: 100
8. ogrenci:
    ad: Travis
    soyad: Hugo
    numara: 44444444
    puan: 90
9. ogrenci:
    ad: Yann
    soyad: LeCun
    numara: 17011020
    puan: 100
10. ogrenci:
    ad: elif
    soyad: gulerarslan
    numara: 12345678
    puan: 100
```

#### iv) Kitap Ödünç Alma

Kitap ödünç alma fonksiyonu olması gerektiği gibi çalışmıyor.

```
void kitapOduncAlma(ODUNCALMA** h, KITAP** head){
    char ID[10];
    char etiket[20];
    ODUNCALMA* yeni =(ODUNCALMA*)malloc(sizeof(ODUNCALMA));
    KITAPODUNC* k =(KITAPODUNC*)malloc(sizeof(KITAPODUNC));
    if(head==NULL || k==NULL){
        return;
    }
    printf("kitabi alacak ogrencinin numarasini giriniz:\n");
    scanf("%s",ID);
    raftakiKitaplar(head);
    printf("almak istediginiz kitabın etiket no'sunu giriniz:\n");
    scanf("%s", etiket);
    printf("odunc alinan gunun tarihini gg.aa.yyyy olacak sekilde giriniz:\n");
    scanf("%d.%d.%d", k->islemTarihi.gun,k->islemTarihi.ay,k->islemTarihi.yil);

    strcpy(k->EtiketNo,etiket);
    strcpy(k->ogrID,ID);

    k->islemTipi=0;
    yeni->p=k;

    kitapOduncGuncel(head, etiket, ID);
}
```

```
void kitapOduncGuncel(KITAP**head, char etiket[20],char* str){
    KITAP* tmp=(KITAP*)malloc(sizeof(KITAP));
    KITAPORNEK* p=(KITAPORNEK*)malloc(sizeof(KITAPORNEK));
    tmp=*head;

    while(tmp!=NULL){
        p=tmp->head;
        while( p!=NULL){
            if(strcmp(etiket,p->EtiketNo)){
                p->next;
            }else{
                strcpy(p->Durum,str);
            }
        }
        tmp=tmp->next;
    }
}
```

## c) Kitap İşlemleri

### i) Raftaki Kitapları Listele

Main fonksiyonunun en başında dosya işlemlerinde dosyadaki bilgilere göre kitaplar linked listi oluşur ve o linked list üzerinden raftaki kitapları ve onların örnekleri görüntülenir.

Yapmak istediğiniz işlemin numarasını giriniz:

14

rafta olan kitaplar:

kitap adi: Java programlama

rafta olan renkleri:

1234567891011\_1

1234567891011\_2

1234567891011\_3

1234567891011\_4

1234567891011\_5

kitap adi: Algoritma Analizi

rafta olan renkleri:

9780140449334\_1

9780140449334\_2

kitap adi: Veri Yapı-lar-ı

rafta olan renkleri:

9780140449335\_1

9780140449335\_2

9780140449335\_3

kitap adi: Fizik 1

rafta olan renkleri:

1234567891012\_1

1234567891012\_2

1234567891012\_3

1234567891012\_4

kitap adi: Fizik 2

rafta olan renkleri:

1234567891013\_1

1234567891013\_2

1234567891013\_3

Java programlama,1234567891011,5

Algoritma Analizi,9780140449334,2

Veri Yapı-lar-ı,9780140449335,3

Fizik 1,1234567891012,4

Fizik 2,1234567891013,3



## ii)Kitap-Yazar Eşleştirme

Dosyalardaki bilgilere göre kitapların yazarlarla birleştirilmesi.

```
yapmak istediginiz islemin numarasini giriniz:
16
Java programlama,Stephen Boyd
Algoritma Analizi,Kalyanmoy Deb
Fizik 1,Michael Jordan
Fizik 1,Danny Leith
Fizik 2,Michael Jordan
Fizik 2,Scott Kirkpatrick
Veri Yapı-lar-,Lieven Vandenberghe
Veri Yapı-lar-,Fabian Pedregosa
Veri Yapı-lar-,Jorge Nocedal
```

Yazarlar:

|                       |        |  |
|-----------------------|--------|--|
| 1,Michael,            | Jordan |  |
| 2,Stephen,Boyd        |        |  |
| 3,Kalyanmoy,Deb       |        |  |
| 4,Danny,Leith         |        |  |
| 5,Scott,Kirkpatrick   |        |  |
| 6,Lieven,Vandenberghe |        |  |
| 7,Fabian,Pedregosa    |        |  |
| 8,Jorge,Nocedal       |        |  |
|                       |        |  |

Kitap Yazar dosyası:

|          |  |
|----------|--|
| 1,23E+12 |  |
| 9,78E+12 |  |
| 1,23E+12 |  |
| 1,23E+12 |  |
| 1,23E+12 |  |
| 1,23E+12 |  |
| 9,78E+12 |  |
| 9,78E+12 |  |
| 9,78E+12 |  |
|          |  |

## 5)Dosya Yazma İşlemleri

```
void dosyaYazdirOgrenci(OGRENCI* head, char* file_name){  
    FILE* fp=fopen(file_name,"w");  
    OGRENCI* tmp=(OGRENCI*)malloc(sizeof(OGRENCI));  
    char buf[10];  
  
    tmp=head;  
    while(tmp!=NULL){  
        fputs(tmp->ogrID,fp);  
        fputc(',',fp);  
        fputs(tmp->ad,fp);  
        fputc(',',fp);  
        fputs(tmp->soyad,fp);  
        fputc(',',fp);  
        itoa(tmp->puan,buf,10);  
        fputs(buf,fp);  
        fputc('\n',fp);  
        tmp=tmp->next;  
    }  
    fclose(fp);  
    return;  
}
```

```
void dosyaYazdirKitapYazar(KITYAZ* head, char* file_name){  
    FILE* fp=fopen(file_name,"w");  
    KITYAZ* tmp=(KITYAZ*)malloc(sizeof(KITYAZ));  
    char buf[10];  
  
    tmp=head;  
    while(tmp!=NULL){  
        fputs(tmp->data->ISBN,fp);  
        fputc(',',fp);  
        itoa(tmp->data->YazarID,buf,10);  
        fputs(buf,fp);  
        fputc('\n',fp);  
        tmp=tmp->next;  
    }  
    fclose(fp);  
    return;  
}
```

```

void dosyaYazdirYazar(YAZAR* head, char* file_name){
    FILE* fp=fopen(file_name,"w");
    YAZAR* tmp=(YAZAR*)malloc(sizeof(YAZAR));
    char buf[10];

    tmp=head;
    while(tmp!=NULL){
        itoa(tmp->yazarID,buf,10);
        fputs(buf,fp);
        fputc(',',fp);
        fputs(tmp->yazarAd,fp);
        fputc(',',fp);
        fputs(tmp->yazarSoyad,fp);
        fputc('\n',fp);
        tmp=tmp->next;
    }
    fclose(fp);
    return;
}

```

## 5)Kazanımlar

Bu proje sayesinde, tam olarak bitirememiş olsam da, dönem boyunca C dili hakkında öğrendiğimiz şeyleri pekiştirdim. Dosya okuma ve yazma işlemlerinin nasıl yapıldığı konusunda daha çok bilgi edindim; linked listlerle işlemlerde, eleman ekleme silmede; dinamik olarak hafıza ayarlama konularında kendimi geliştirdim.

## 6)Kaynakça

-<https://stackoverflow.com/questions/6417158/c-how-to-free-nodes-in-the-linked-list>

-[https://www.w3resource.com/c-programming-exercises/linked\\_list/c-linked\\_list-exercise-15.php](https://www.w3resource.com/c-programming-exercises/linked_list/c-linked_list-exercise-15.php)

-Ders materyalleri