

```
struct Student{
    int ogr_no;
    char ad[20];
    char soyad[20];
    char bolum[40];
} typedef student;
```

Yanda görüldüğü üzere öğrenci kayıtlarını sabit uzunluklu olarak dosyada tutabilmek için struct yapısı oluşturdum.

```
int main(){
    FILE *students;
    student blank = {-1, "", "", ""};

    students = fopen("students.bin", "rb+");
    if (!students)
        return 0;
    else{
        map<int, int> index = getIndexMap(students, blank);
        int choise = getChoise();

        switch (choise){
            case 1:
                ogrenciEkle_1(students, blank);
                break;
            case 2:
                ogrenciSil_2(students, blank);
                break;
            case 3:
                ogrenciGuncelle_3(students, index, blank);
                break;
            case 4:
                ogrenciGoruntule_4(students, index, blank);
                break;
        }
    }
    fclose(students);
}
```

Görüldüğü gibi ilk başta binary dosyamı açtım, yoksa da oluşturdum. Sonrasında dosyayı parametre olarak gönderip indexleme işlemi yaptım. Son olarak da kullanıcıdan seçim alıp ona göre dosya işlemi gerçekleştirdim.

```

map<int, int> getIndexMap(FILE *f, student &blank){
    map<int, int> indexMap;

    int i = 1;

    fread(&blank, sizeof(student), 1, f);
    while (!feof(f))
        indexMap[blank.ogr_no] = i++, fread(&blank, sizeof(student), 1, f);
    rewind(f);
    return (indexMap);
}

int getChoise(){
    int a;

    printf("[1] Öğrenci Ekle\n"
           "[2] Öğrenci Sil\n"
           "[3] Öğrenci Güncelle\n"
           "[4] Öğrenci Kaydını Görüntüle\n"
           "Seçiminiz >> ");
    scanf("%d", &a);
    while (!(a >= 1 && a <= 4))
    {
        fflush(stdout);
        printf("1-4 arasından seçim giriniz >> ");
        scanf("%d", &a);
    }
    return (a);
}

```

Görüldüğü üzere dosyayı baştan sona okuyarak öğrenci noya göre map yapımda kaydediyorum.

getChoise() fonksiyonu ile ise kullanıcıdan girdi alıyorum.

```

void ogrenciEkle_1(FILE *f, student& blank){
    fflush(stdout);
    printf("Öğrenci No: ");
    scanf("%d", &(blank.ogr_no));
    fflush(stdout);
    printf("Öğrenci Ad: ");
    scanf("%s", blank.ad);
    fflush(stdout);
    printf("Öğrenci Soyad: ");
    scanf("%s", &(blank.soyad));
    fflush(stdout);
    printf("Öğrenci Bölüm: ");
    scanf("%s", blank.bolum);

    fseek(f, 0, SEEK_END);
    fwrite(&blank, sizeof(student), 1, f);
}

```

Yanda görüldüğü gibi öğrenci eklemek için sırayla gerekli girdileri alıp fwrite fonksiyonum ile dosya sonuna ekliyorum.

```

void ogrenciSil_2(FILE *f, student& blank){
    int a;

    printf("Öğrenci No >> ");
    scanf("%d", &a);

    FILE *tmp_f = fopen("tmp.bin", "wb");
    fread(&blank, sizeof(student), 1, f);
    while(!feof(f)){
        if (blank.ogr_no != a){
            fwrite(&blank, sizeof(student), 1, tmp_f);
        }
        fread(&blank, sizeof(student), 1, f);
    }
    fclose(tmp_f);
    remove("students.bin");
    rename("tmp.bin", "students.bin");
}

```

Öğrenci silmek için geçici dosya oluşturup spesifik kayıt olmadan dosyayı yazıyorum ve ismini değiştiriyorum.

Normalde o record' u null atayıp liste ile boş kayıtları tutabilirdim ama böyle daha kolay geldi.

```

void ogrenciGuncelle_3(FILE *f, map<int, int>& index, student &blank){
    int a;

    printf("Öğrenci No >> ");
    scanf("%d", &a);
    if (index[a] == 0){
        printf("Öğrenci Bulunamadı");
        return ;
    }
    fseek(f, sizeof(student) * (index[a] - 1), SEEK_SET);
    fflush(stdout);
    printf("\nÖğrenci No: ");
    scanf("%d", &(blank.ogr_no));
    fflush(stdout);
    printf("Öğrenci Ad: ");
    scanf("%s", blank.ad);
    fflush(stdout);
    printf("Öğrenci Soyad: ");
    scanf("%s", &(blank.soyad));
    fflush(stdout);
    printf("Öğrenci Bölüm: ");
    scanf("%s", blank.bolum);
    fwrite(&blank, sizeof(student), 1, f);
}

```

Öğrenci güncellemek için doğrudan kayıt' ın bulunduğu index' e gidip aynı boyutta yeni bir kayıt ile üstüne yazıyorum.

```

void ogrenciGoruntule_4(FILE *f, map<int, int>& index, student& blank){
    int a;

    printf("[1] Tek Öğrenci Görüntüle\n"
           "[2] Tüm Öğrencileri Görüntüle\n"
           "Seçiminiz >> ");
    scanf("%d", &a);
    while (!(a >= 1 && a <= 2))
    {
        fflush(stdout);
        printf("1-2 aras[] seçim giriniz >> ");
        scanf("%d", &a);
    }

    switch(a){
        case 1:
            printf("Öğrenci No >> ");
            scanf("%d", &a);
            if (index[a] == 0){
                printf("Öğrenci Bulunamad[]");
                return ;
            }
            fseek(f, sizeof(student)* (index[a] - 1), SEEK_SET);
            fread(&blank, sizeof(student), 1, f);
            printf("Öğrenci No: %d\nİsim: %s %s\nBölüm: %s\n", blank.ogr_no, blank.ad, blank.soyad, blank.bolum);
            break;
        case 2:
            rewind(f);
            fread(&blank, sizeof(student), 1, f);
            while(!feof(f)){
                printf("Öğrenci No: %d\nİsim: %s %s\nBölüm: %s\n\n", blank.ogr_no, blank.ad, blank.soyad, blank.bolum);
                fread(&blank, sizeof(student), 1, f);
            }
            break;
    }
}

```

Öğrenci görüntüleme için iki seçenek ekledim ya tüm öğrencileri görüntüleyecek ya da öğrenci no verip spesifik indexte bulunan tek öğrenciyi görüntüleyecek.