Ahmet Saraç 1810206065

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

struct ogrenci {
   int no;
   char isim[40];
   int vize, final;
   struct ogrenci *sonraki;
};

typedef struct ogrenci dugum;

dugum *head, *yeniDugum, *enBasarili, *p, *q;

void kayitEkle() {
   int kayitSayisi = 0;
   printf("Kac ogrenci gireceksiniz : ");
   scanf("%d", &kayitSayisi);
   for (int i = 0; i < kayitSayisi; i++) {
      yeniDugum = (dugum *)malloc(sizeof(dugum));
      p->sonraki = yeniDugum;
      printf("Ogrenci No : ");
      scanf("%d", &yeniDugum->no);
      printf("Ogrenci Isim : ");
      scanf("%s", yeniDugum->isim);
      printf("Ogrenci Vize Notu : ");
      scanf("%d", &yeniDugum->vize);
      printf("Ogrenci Final Notu : ");
```

```
        scanf("%d", &yeniDugum->final);
        p = p->sonraki;
        yeniDugum = yeniDugum->sonraki;
    }
    p->sonraki = NULL;
}

float notHesapla(int vize, int final) {
    return (vize * 0.4) + (final * 0.6);
}

void kayitListele(int altLimit) {
    q = head->sonraki;
    while (q != NULL) {
        if (notHesapla(q->vize, q->final) >= altLimit) {
            printf("Ogrenci No : %d\t", q->no);
            printf("Ogrenci Ad : %s\t", q->isim);
            printf("Ogrenci Vize : %d\t", q->vize);
            printf("Ogrenci Final : %d\t", q->final);
            printf("Ortalama Not : %.2f\t\n", notHesapla(q->vize, q->final));
        }
        q = q->sonraki;
    }
}

void kayitGuncelle(int no) {
    int secim;
    q = head->sonraki;
    while (q != NULL) {
        if (q->no == no) {
            while (1) {
```

```
            printf("Guncellemek istediginiz secenegi giriniz : No[1]\t Isim[2]\t Vize Notu[3]\t Final
Notu[4] Ust Menu icin [0]: ");
            scanf("%d", &secim);
            switch (secim) {
            case 1:
               printf("Guncellenecek no degerini giriniz : ");
               scanf("%d", &q->no);
               break;
            case 2:
               printf("Guncellenecek isimi giriniz : ");
               scanf("%s", q->isim);
               break;
            case 3:
               printf("Guncellenecek vize notunu giriniz : ");
               scanf("%d", &q->vize);
               break;
            case 4:
               printf("Guncellecek final notunu giriniz : ");
               scanf("%d", &q->final);
               break;
            }
            if (secim == 0) {
               break;
            }
         }
         break;
      }
      q = q->sonraki;
   }
}
```

```
void ortalamaHesapla() {
    q = head->sonraki;
    float toplam = 0;
    int sayac = 0;
    if (q == NULL) {
        printf("Listede kimse yok");
    } else {
        for (; q != NULL; sayac++, q = q->sonraki) {
            toplam += notHesapla(q->vize, q->final);
        }
        printf("Sinif ortalamasi : %.2f\n", toplam / sayac);
    }
}

void enBasariliOgrenciGoster() {

    q = head->sonraki;
    enBasarili = head->sonraki;

    while (q->sonraki != NULL) {
        q = q->sonraki;
        if (notHesapla(q->vize, q->final) > notHesapla(enBasarili->vize, enBasarili->final)) {
            enBasarili = q;
        }
    }

    printf("En basarili ogrenci :\n");
    printf("No : %d\t Isim : %s\t Vize : %d\t Final : %d\t Ortalama : %.2f\n", enBasarili->no, enBasarili->isim, enBasarili->vize, enBasarili->final, notHesapla(enBasarili->vize, enBasarili->final));
}
```

```
int main() {


    int secim = 0;

    int altLimit = 0;

    int guncellemeNo = 0;

    printf("1. Yeni Kayit Ekleme\n2. Kayit Listeleme\n3. Kayit Guncelleme\n4. Sinif Ortalamasi
Hesapla\n5. Ortalamaya Gore En Basarili Ogrenci Bilgisini Goster\n");

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

    p = head;

    head->sonraki = NULL;

    while (1) {

        printf("Secim Yap [1 - 5] [Programdan cikmak icin 0'a bas] : ");

        scanf("%d", &secim);

        switch (secim) {

        case 1:

            kayitEkle();

            break;

        case 2:

            printf("Ortalama alt limiti giriniz :");

            scanf("%d", &altLimit);

            kayitListele(altLimit);

            break;

        case 3:

            printf("Kaydini guncellemek istediginiz ogrencinin numarasini giriniz : ");

            scanf("%d", &guncellemeNo);

            kayitGuncelle(guncellemeNo);

            break;

        case 4:

            ortalamaHesapla();

            break;

        case 5:
```

```
            enBasariliOgrenciGoster();
            break;
        case 0:
            exit(0);
            break;
        }
    }

    getch();
    return 0;
}
```