

```

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

enum Sinif {Bir = 1, Iki = 2, Uc = 3, Dort = 4};

#define birinci_donem_ders_sayisi 7
#define ikinci_donem_ders_sayisi 7

struct ders {
    char kodu[50];
    double kredisi;
    char * adi;
};

typedef struct ders Ders;

struct birinci_donem {
    float ders_sayisi;
    float ortalama;
    float toplam_kredi;
    Ders* dersler[];
};

typedef struct birinci_donem Birinci_Donem;

struct ikinci_donem {
    float ders_sayisi;
    float ortalama;
    float toplam_kredi;
    Ders *dersler[];
};

typedef struct ikinci_donem Ikinci_Donem;

extern char menu(int);
extern void ek(float);
extern char devam_mi(void);
extern void printf_ders(Ders ders);
extern void printf_ders_ptr(Ders *ders);
extern void* yeni_birinci_donem();
extern void yeni_birinci_donem_ptr(Birinci_Donem *donem);
extern void* yeni_ikinci_donem();
extern void yeni_ikinci_donem_ptr(Ikinci_Donem *donem);
extern void birinci_sinif_not_hesabi();
extern int main();

extern char menu(int i)
{
    printf("\n*****MENU*****\n\n\n");
    printf("    Sadece %d. sinif notlarini hesaplama kullanabilirsiniz!\n",i);
    printf("    1)1. donem not hesabi\n");
    printf("    2)2. donem not hesabi\n");
    printf("    3)Kredilerin görüntülenmesi\n");
    printf("    4)Cikis\n");
    printf("    Seciminiz[1-4]\n?");
    fflush(stdin);
    return getchar();
}

extern void ek(float ortalama)
{
    printf("Diger donem notunuz en az %5.2f olmalı",ortalama);
    printf("\nyada yaz okulundan ders almalısınız");
    printf("\nIsterseniz notlarınızı yükseltmek");
    printf("\nicinde yaz okulundan ders alabilirsiniz");
}

extern char devam_mi(void)
{

```

```

printf("\n0)Menu\n?");
printf("\n1)Devam\n?");
fflush(stdin);
return getchar();
}
extern void printf_ders(Ders ders)
{
    printf("%-13s:%2.0f\n",ders.adi,ders.kredisi);
}
extern void printf_ders_ptr(Ders *ders)
{
    printf("%-13s:%2.0f\n",ders->adi,ders->kredisi);
}
extern void* printf_birinci_donem_ptr_kredisi(Birinci_Donem *birinci_donem)
{
    return NULL;
}
extern void sistem_rengini_ayarla()
{
    char *renk= {"color A"};
    system(renk);
}
extern void* yeni_birinci_donem()
{
    return malloc(sizeof(Birinci_Donem));
}
extern void yeni_birinci_donem_ptr(Birinci_Donem *donem)
{
    donem=yeni_birinci_donem();

//donem=malloc(sizeof(Birinci_Donem));
}
extern void* yeni_ikinci_donem()
{
    return malloc(sizeof(Ikinci_Donem));
}
extern void yeni_ikinci_donem_ptr(Ikinci_Donem *donem)
{
    donem=yeni_ikinci_donem();
}
extern void birinci_sinif_not_hesabi()
{
    int i=0;

    char secim;
    char *s=" notunu giriniz\n?";
    char *birinci_donem_dersleri[birinci_donem_ders_sayisi]= {"Algoritma ve
Programlama-I","Matematik I","Fizik I","Ingilizce","Bilgisayar Muhendisligine Giris",
"Edebiyat I","Tarih I"};
    char *ikinci_donem_dersleri[ikinci_donem_ders_sayisi]= {"Algoritma ve Programlama-II",
"Matematik II","Fizik II","Ayrik Yapilar","Bilgisayar Bilimlerine Giris","Edebiyat II",
"Tarih II"};

    float birinci_donem_kredileri[birinci_donem_ders_sayisi]= {4,3,4,3,1,2,2};
    float ikinci_donem_kredileri[ikinci_donem_ders_sayisi]= {4,3,4,3,3,2,2};
    float toplam=0;
    float puan=0;

    Birinci_Donem *birinci_donem=NULL;
    Ikinci_Donem *ikinci_donem=NULL;

    do {
        birinci_donem=yeni_birinci_donem();
    } while(birinci_donem==NULL);

    do {

```

```

        ikinci_donem=yeni_ikinci_donem();
    } while(ikinci_donem==NULL);

    birinci_donem->ders_sayisi=birinci_donem_ders_sayisi;
    ikinci_donem->ders_sayisi=ikinci_donem_ders_sayisi;

    for(i=0; i<birinci_donem->ders_sayisi; i++) {
        do {
            birinci_donem->dersler[i]=malloc(sizeof(Ders));
        } while(birinci_donem->dersler[i]==NULL);
        birinci_donem->dersler[i]->adi=birinci_donem_dersleri[i];
        printf("%s",birinci_donem->dersler[i]->adi);
        birinci_donem->toplam_kredi+=birinci_donem->dersler[i]->kredisi=
    birinci_donem_kredileri[i];
    }
    for(i=0; i<ikinci_donem->ders_sayisi; i++) {
        do {
            ikinci_donem->dersler[i]=malloc(sizeof(Ders));
        } while(ikinci_donem->dersler[i]==NULL);
        ikinci_donem->dersler[i]->adi=ikinci_donem_dersleri[i];
        ikinci_donem->toplam_kredi+=ikinci_donem->dersler[i]->kredisi=
    ikinci_donem_kredileri[i];
    }

    do {
        fflush(stdin);
        secim=menu(Bir);
        switch (secim) {
            case '1':
                do {
                    for (i=0; i<birinci_donem->ders_sayisi; i++) {
                        //printf("%s",birinci_donem->dersler[i]->adi,s);
                        fflush(stdin);
                        scanf("%f",&puan);
                        toplam+=puan*birinci_donem->dersler[i]->kredisi;
                    }
                    birinci_donem->ortalama=toplam/birinci_donem->toplam_kredi;
                    toplam=0;
                    printf("1. donem agnonuz:%5.2f\n",birinci_donem->ortalama);
                    ek(3.6-birinci_donem->ortalama);
                } while (devam_mi()=="1");
                break;
            case '2':
                do {
                    for (i=0; i<ikinci_donem->ders_sayisi; i++) {
                        printf("%s",ikinci_donem->dersler[i]->adi,s);
                        fflush(stdin);
                        scanf("%f",&puan);
                        toplam+=puan*ikinci_donem->dersler[i]->kredisi;
                    }
                    ikinci_donem->ortalama=toplam/ikinci_donem->toplam_kredi;
                    toplam=0;
                    printf("2. donem agnonuz:%5.2f\n",ikinci_donem->ortalama);
                    ek(3.6-ikinci_donem->ortalama);
                } while (devam_mi()=="1");
                break;
            case '3':
                printf("\nBirinci donem kredileri:\n\n");
                //printf_birinci_donem_ptr_kredisi();
                for (i=0; i<birinci_donem->ders_sayisi; i++) {
                    printf_ders_ptr(birinci_donem->dersler[i]);
                }
                printf("\nIkinci donem kredileri:\n\n");
                for (i=0; i<ikinci_donem->ders_sayisi; i++) {
                    printf_ders_ptr(ikinci_donem->dersler[i]);
                }
            }
        }
    }

```

```
        }
        printf("\n");
        break;
    default:
        secim='4';
        break;
    }
} while (secim!='4');

}
extern int main()
{
    sistem_rengini_ayarla();
    birinci_sinif_not_hesabi();
    return EXIT_SUCCESS;
}
```