

Struct



```
struct number {  
    int img;  
    float real;  
};
```

Gömülü Sistemler Laboratuvarı

Fonksiyon & Struct

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

```
#include <math.h>
```

```
struct Nokta {
```

```
    int x;
```

```
    int y;
```

```
};
```

```
int main() {
```

```
}
```

Fonksiyon & Struct

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

```
#include <math.h>
```

```
struct Nokta {
```

```
    int x;
```

```
    int y;
```

```
};
```

```
int main() {
```

```
    struct Nokta nokta1 = {1, 4};
```

```
    struct Nokta nokta2 = {1, 2};
```

```
    printf("nokta1 x,y : %d,%d\n", nokta1.x, nokta1.y);
```

```
    printf("nokta2 x,y : %d,%d\n\n", nokta2.x, nokta2.y);
```

```
    return 0;
```

```
}
```

Fonksiyon & Struct

```
#include <stdio.h>
#include <math.h>
struct Nokta {
    int x;
    int y;
};

double Nokta_uzaklik(struct Nokta n1, struct Nokta n2) {
    return sqrt(pow(n1.x-n2.x, 2) + pow(n1.y-n2.y, 2));
}

int main() {
    struct Nokta nokta1 = {1, 4};
    struct Nokta nokta2 = {1, 2};

    printf("nokta1 x,y : %d,%d\n", nokta1.x, nokta1.y);
    printf("nokta2 x,y : %d,%d\n", nokta2.x, nokta2.y);

    return 0;
}
```

Fonksiyon & Struct

```
#include <stdio.h>
#include <math.h>
struct Nokta {
    int x;
    int y;
};

double Nokta_uzaklik(struct Nokta n1, struct Nokta n2) {
    return sqrt(pow(n1.x-n2.x, 2) + pow(n1.y-n2.y, 2));
}

int main() {
    struct Nokta nokta1 = {1, 4};
    struct Nokta nokta2 = {1, 2};

    printf("nokta1 x,y : %d,%d\n", nokta1.x, nokta1.y);
    printf("nokta2 x,y : %d,%d\n", nokta2.x, nokta2.y);

    double uzaklik = Nokta_uzaklik(nokta1, nokta2);
    printf("nokta1 ve nokta2 arasindaki uzaklik: %f\n", uzaklik);

    return 0;
}
```

Fonksiyon & Struct

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

```
#include <math.h>
```

```
struct Nokta {
```

```
    int x;
```

```
    int y;
```

```
};
```

```
int main() {
```

```
    struct Nokta nokta1 = {1, 4};
```

```
    return 0;
```

```
}
```

Fonksiyon & Struct

```
#include <stdio.h>
#include <math.h>
struct Nokta {
    int x;
    int y;
};

void Nokta_deger_ata_1(struct Nokta nokta, int x, int y) {
    nokta.x = x;
    nokta.y = y;
}

int main() {
    struct Nokta nokta1 = {1, 4};

    Nokta_deger_ata_1(nokta1, 100, 200);

    return 0;
}
```

Fonksiyon & Struct

```
#include <stdio.h>
#include <math.h>
struct Nokta {
    int x;
    int y;
};

void Nokta_deger_ata_1(struct Nokta nokta, int x, int y) {
    nokta.x = x;
    nokta.y = y;
}

int main() {
    struct Nokta nokta1 = {1, 4};

    Nokta_deger_ata_1(nokta1, 100, 200);
    printf("deger_ata_1 fonksiyonu sonrasinda\n");
    printf("nokta1 x,y : %d,%d\n\n", nokta1.x, nokta1.y);

    return 0;
}
```


Fonksiyon & Struct

```
#include <stdio.h>
#include <math.h>
struct Nokta {
    int x;
    int y;
};
```

```
int main() {
    struct Nokta nokta1 = {1, 4};
    struct Nokta nokta2 = {1, 2};

    return 0;
}
```

Fonksiyon & Struct

```
#include <stdio.h>
#include <math.h>
struct Nokta {
    int x;
    int y;
};

void Nokta_deger_ata_2(struct Nokta *nokta, int x, int y) {
    (*nokta).x = x;
    (*nokta).y = y;
}

int main() {
    struct Nokta nokta1 = {1, 4};
    struct Nokta nokta2 = {1, 2};
    Nokta_deger_ata_2(&nokta1, 2, 1);
    printf("deger_ata_2 fonksiyonu sonrasinda\n");
    printf("nokta1 x,y : %d,%d\n\n", nokta1.x, nokta1.y);

    return 0;
}
```

Fonksiyon & Struct

```
#include <stdio.h>
#include <math.h>
struct Nokta {
    int x;
    int y;
};
void Nokta_deger_ata_2(struct Nokta *nokta, int x, int y) {
    (*nokta).x = x;
    (*nokta).y = y;
}
double Nokta_uzaklik_2(struct Nokta *n1, struct Nokta *n2) {
    return sqrt(pow(n1->x - n2->x, 2) + pow(n1->y - n2->y, 2));
}
int main() {
    struct Nokta nokta1 = {1, 4};
    struct Nokta nokta2 = {1, 2};
    Nokta_deger_ata_2(&nokta1, 2, 1);
    printf("deger_ata_2 fonksiyonu sonrasinda\n");
    printf("nokta1 x,y : %d,%d\n", nokta1.x, nokta1.y);
    double uzaklik = Nokta_uzaklik_2(&nokta1, &nokta2);
    printf("nokta1 ve nokta2 arasindaki uzaklik: %f\n", uzaklik);
    return 0;
}
```

Struct Tipinde Dizi

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

```
struct Ogrenci {
```

```
};
```

```
int main() {
```

```
    struct Ogrenci ogr1;
```

```
    return 0;
```

```
}
```

Struct Tipinde Dizi

```
#include <stdio.h>
#include <string.h>
struct Tarih {
```

```
    int gun;
```

```
    int ay;
```

```
    int yil;
```

```
};
```

```
struct Ogrenci {
```

```
    char ad[21];
```

```
    char soyad[21];
```

```
    char no[11];
```

```
    char bolum[41];
```

```
    struct Tarih kayit_tarihi;
```

```
};
```

```
int main() {
```

```
    struct Ogrenci ogr1;
```

```
    return 0;
```

```
}
```

Struct Tipinde Dizi

```
#include <stdio.h>
#include <string.h>
struct Tarih {
    int gun;
    int ay;
    int yıl;
};

struct Ogrenci {
    char ad[21];
    char soyad[21];
    char no[11];
    char bolum[41];
    struct Tarih kayit_tarihi;
};

void Ogrenci_atama_yap(struct Ogrenci *ogr, const char adi[ ], const char soyadi[ ], const char numarası[ ], const char bolumu[ ], struct Tarih kayit_oldugu_tarih) {
    strcpy(ogr->ad, adi);
    strcpy(ogr->soyad, soyadi);
    strcpy(ogr->no, numarası);
    strcpy(ogr->bolum, bolumu);
    ogr->kayit_tarihi = kayit_oldugu_tarih;
}

int main() {
    struct Ogrenci ogr1;
    Ogrenci_atama_yap(&ogr1, "Abcd", "Soyad", "150202175", "Bilgisayar Muhendisligi", Tarih_olustur(01, 02, 2015));

    return 0;
}
```

Struct Tipinde Dizi

```
#include <stdio.h>
#include <string.h>
struct Tarih {
    int gun;
    int ay;
    int yil;
};
struct Tarih Tarih_olustur(int gun, int ay, int yil) {
    struct Tarih t;
    if (gun > 31 || ay > 12) {
        t.gun = 0;
        t.ay = 0;
        t.yil = 0;
    } else {
        t.gun = gun;
        t.ay = ay;
        t.yil = yil;
    }
    return t;
}
struct Ogrenci {
    char ad[21];
    char soyad[21];
    char no[11];
    char bolum[41];
    struct Tarih kayit_tarihi;
};
void Ogrenci_atama_yap(struct Ogrenci *ogr, const char adi[], const char soyadi[], const char numarasi[], const char bolumu[], struct Tarih kayit_oldugu_tarih) {
    strcpy(ogr->ad, adi);
    strcpy(ogr->soyad, soyadi);
    strcpy(ogr->no, numarasi);
    strcpy(ogr->bolum, bolumu);
    ogr->kayit_tarihi = kayit_oldugu_tarih;
}
int main() {
    struct Ogrenci ogr1;
    Ogrenci_atama_yap(&ogr1, "Abcd", "Soyad", "150202175", "Bilgisayar Muhendisligi", Tarih_olustur(01, 02, 2015));

    return 0;
}
```

Struct Tipinde Dizi

```
#include <stdio.h>
#include <string.h>
struct Tarih {
    int gun;
    int ay;
    int yil;
};
struct Tarih Tarih_olustur(int gun, int ay, int yil) {
    struct Tarih t;
    if (gun > 31 || ay > 12) {
        t.gun = 0;
        t.ay = 0;
        t.yil = 0;
    } else {
        t.gun = gun;
        t.ay = ay;
        t.yil = yil;
    }
    return t;
}
struct Ogrenci {
    char ad[21];
    char soyad[21];
    char no[11];
    char bolum[41];
    struct Tarih kayit_tarihi;
};
void Ogrenci_atama_yap(struct Ogrenci *ogr, const char adi[], const char soyadi[], const char numarasi[], const char bolumu[], struct Tarih kayit_oldugu_tarih) {
    strcpy(ogr->ad, adi);
    strcpy(ogr->soyad, soyadi);
    strcpy(ogr->no, numarasi);
    strcpy(ogr->bolum, bolumu);
    ogr->kayit_tarihi = kayit_oldugu_tarih;
}
int main() {
    struct Ogrenci ogr1;
    Ogrenci_atama_yap(&ogr1, "Abcd", "Soyad", "150202175", "Bilgisayar Muhendisligi", Tarih_olustur(01, 02, 2015));
    printf("adi: %s\n", ogr1.ad);
    printf("soyadi: %s\n", ogr1.soyad);
    printf("no: %s\n", ogr1.no);
    printf("bolum: %s\n", ogr1.bolum);
    printf("kayit yili: %d\n\n", ogr1.kayit_tarihi.yil);
    return 0;
}
```


Struct Tipinde Dizi

```
#include <stdio.h>
#include <string.h>
struct Tarih {
    int gun;
    int ay;
    int yil;
};

struct Tarih Tarih_olustur(int gun, int ay, int yil) {
    struct Tarih t;
    if (gun > 31 || ay > 12) {
        t.gun = 0;
        t.gun = gun;
        t.ay = 0;
        t.ay = ay;
        t.yil = 0;
        t.yil = yil;
    }
    return t;
}

struct Ogrenci {
    char ad[21];
    char soyad[21];
    char no[11];
    char bolum[41];
    struct Tarih kayit_tarihi;
};

void Ogrenci_atama_yap(struct Ogrenci *ogr, const char adi[], const char soyadi[], const char numarası[], const char bolumu[], struct Tarih kayit_oldugu_tarih) {
    strcpy(ogr->ad, adi);
    strcpy(ogr->soyad, soyadi);
    strcpy(ogr->no, numarası);
    strcpy(ogr->bolum, bolumu);
    ogr->kayit_tarihi = kayit_oldugu_tarih;
}

int main() {
    struct Ogrenci ogr_liste[20];

    return 0;
}
```

Struct Tipinde Dizi

```
#include <stdio.h>
#include <string.h>
struct Tarih {
    int gun;
    int ay;
    int yil;
};

struct Tarih Tarih_olustur(int gun, int ay, int yil) {
    struct Tarih t;
    if (gun > 31 || ay > 12) {
        t.gun = 0;
        t.gun = gun;
        t.ay = 0;
        t.ay = ay;
        t.yil = 0;
        t.yil = yil;
    }
    return t;
}

struct Ogrenci {
    char ad[21];
    char soyad[21];
    char no[11];
    char bolum[41];
    struct Tarih kayit_tarihi;
};

void Ogrenci_atama_yap(struct Ogrenci *ogr, const char adi[], const char soyadi[], const char numarasi[], const char bolumu[], struct Tarih kayit_oldugu_tarih) {
    strcpy(ogr->ad, adi);
    strcpy(ogr->soyad, soyadi);
    strcpy(ogr->no, numarasi);
    strcpy(ogr->bolum, bolumu);
    ogr->kayit_tarihi = kayit_oldugu_tarih;
}

int main() {
    struct Ogrenci ogr_liste[20];
    Ogrenci_atama_yap(&ogr_liste[0], "A", "A", "150202177", "BilgisayarMuhendisligi", Tarih_olustur(1, 2, 2015));
    Ogrenci_atama_yap(&ogr_liste[1], "B", "B", "150202178", "BilgisayarMuhendisligi", Tarih_olustur(1, 2, 2015));
    Ogrenci_atama_yap(&ogr_liste[2], "C", "C", "150202179", "BilgisayarMuhendisligi", Tarih_olustur(2, 9, 2014));

    return 0;
}
```

Struct Tipinde Dizi

```
#include <stdio.h>
#include <string.h>
struct Tarih {
    int gun;
    int ay;
    int yil;
};
struct Tarih Tarih_olustur(int gun, int ay, int yil) {
    struct Tarih t;
    if (gun > 31 || ay > 12) {
        t.gun = 0;
        t.gun = gun;
        t.ay = 0;
        t.ay = ay;
        t.yil = 0;
        t.yil = yil;
    }
    return t;
}
struct Ogrenci {
    char ad[21];
    char soyad[21];
    char no[11];
    char bolum[41];
    struct Tarih kayit_tarihi;
};
void Ogrenci_atama_yap(struct Ogrenci *ogr, const char adi[], const char soyadi[], const char numarası[], const char bolumu[], struct Tarih kayit_oldugu_tarih) {
    strcpy(ogr->ad, adi);
    strcpy(ogr->soyad, soyadi);
    strcpy(ogr->no, numarası);
    strcpy(ogr->bolum, bolumu);
    ogr->kayit_tarihi = kayit_oldugu_tarih;
}
int main() {
    struct Ogrenci ogr_liste[20];
    Ogrenci_atama_yap(&ogr_liste[0], "A", "A", "150202177", "BilgisayarMuhendisligi", Tarih_olustur(1, 2, 2015));
    Ogrenci_atama_yap(&ogr_liste[1], "B", "B", "150202178", "BilgisayarMuhendisligi", Tarih_olustur(1, 2, 2015));
    Ogrenci_atama_yap(&ogr_liste[2], "C", "C", "150202179", "BilgisayarMuhendisligi", Tarih_olustur(2, 9, 2014));
    int kayitli_ogr_sayisi = 3;
    for (int i = 0; i < kayitli_ogr_sayisi; i++) {
        printf("adi soyad: %s %s\n", ogr_liste[i].ad, ogr_liste[i].soyad);
        printf("no: %s\n", ogr_liste[i].no);
        printf("bolum: %s\n", ogr_liste[i].bolum);
        printf("kayit tarihi: %02d.%02d.%d\n", ogr_liste[i].kayit_tarihi.gun, ogr_liste[i].kayit_tarihi.ay, ogr_liste[i].kayit_tarihi.yil);
    }
    return 0;
}
```

Sorular

