

## Diziler

### - Dizi Tanımlama

```
int sayilar[]={2, 56, 42};
int *p=sayilar;
cout<<*p;
```

### - Dizi Uzunluk

```
int uzunluk;
int uzunluk;
cout<<"Dizi Uzunlugu:";
cin>>uzunluk;
int sayilar[uzunluk];
for(int i=0;i<uzunluk;i++){
sayilar[i]=0;
}
```

### - Dizi Değerlerinin Sıfırlanması

```
int* DiziOlusturSifirla(int uzunluk){
    int sayilar[uzunluk];
    for(int i=0;i<uzunluk;i++){
        sayilar[i]=0;
    }
    return sayilar;
}

int main(){
    int uzunluk;
    cout<<"Dizi Uzunlugu:";
    cin>>uzunluk;
    int *p = DiziOlusturSifirla(uzunluk);
    cout<<p[1];
    return 0;
}
```

### - Heap 'te Dizi Oluşturma

```
int* DiziOlusturSifirla(int uzunluk){
    int *sayilar = new int[uzunluk];
    for(int i=0;i<uzunluk;i++){
        sayilar[i]=0;
    }
    return sayilar;
}

int main(){
    int uzunluk;
    cout<<"Dizi Uzunlugu:";
    cin>>uzunluk;
    int *p = DiziOlusturSifirla(uzunluk);
    cout<<p[1];
    delete p;
    return 0;
}
```

### - Hata Verecek Kod Örneği

```
class Kisi{
    private:
        string isim;
        int yas;
    public:
        Kisi(string ism){
            isim=ism;
            yas=0;
        }
};

int main(){
```

```

        Kisi kisiler[10];
        return 0;
}

```

- Doğru Kullanım

```

#include <iostream>
using namespace std;
class Kisi{
    private:
        string isim;
        int yas;
    public:
        Kisi(string ism){
            isim=ism;
            yas=0;
        }
};
int main(){
    Kisi **kisiler = new Kisi*[10];
    for(int i=0;i<10;i++) kisiler[i] = new Kisi("Mehmet");

    for(int i=0;i<10;i++) delete kisiler[i];
    return 0;
}

```

- C dili tarzı string veri türleri

```

int main(){
    char* str = "Kelaynak";
    cout<<*str<<endl; // sadece K harfini yazar
    cout<<str;        // Bütün bir string'i ekrana yazar.
    return 0;
}

```

- Uzunluğun Bulunması

```

int Uzunluk(char *str){
    int u=0;
    for(int i=0;str[i] != '\0';i++) u++;
    return u;
}

```

- Dizi ataması ve Hatalı kullanım

```

int a[]={3,7,9};
int b[3];
b=a;

```

- Doğru Atama Yöntemi

```

int a[]={3,7,9};
int b[3];
for(int i=0;i<3;i++) b[i]=a[i];

```

- Dizi Karşılaştırma Adres Olarak

```

int a[]={3,7,9};
int b[]={3,7,9};
if(a == b) cout<<"diziler esit";
else cout<<"diziler esit degil";

```

- Dizi Karşılaştırma Değer Olarak

```

bool Esitmi(int d1[],int d2[],int u1,int u2){
    if(u1 != u2) return false;
    for(int i=0;i<u1;i++){
        if(d1[i] != d2[i]) return false;
    }
}

```

```

    }
    return true;
}

int main(){
    int a[]={3,7,9};
    int b[]={3,7,9};
    if(Esitmi(a,b,3,3)) cout<<"diziler esit";
    else cout<<"diziler esit degil";
    return 0;
}

```

## İki Boyutlu Diziler

```

#include <iostream>
#include <ctime>

using namespace std;
const int sutun=3;

void Yazdir(int dizi[][sutun],int satir){
    for(int i=0;i<satir;i++){
        for(int j=0;j<sutun;j++){
            cout<<dizi[i][j]<<" ";
        }
        cout<<endl;
    }
}

int main(){
    srand(time(NULL));
    int sayilar[4][sutun];
    for(int i=0;i<4;i++){
        for(int j=0;j<sutun;j++){
            sayilar[i][j] = rand()%9+1;
        }
    }
    Yazdir(sayilar,4);
    return 0;
}

```

## Heap Örneği

```

#include <iostream>
using namespace std;

class Kisi{
private:
    string isim;
public:
    Kisi(string ism){
        isim=ism;
    }
    friend ostream& operator<<(ostream& ekran,Kisi &sag){
        ekran<<sag.isim<<endl;
    }
}

```

```

        return ekran;
    }
};

void Yazdir(Kisi **kisiDizi, int kisiSayisi){
    for(int i=0;i<kisiSayisi;i++){
        cout<<*kisiDizi[i];
    }
}

void Yoket(Kisi **kisiDizi, int kisiSayisi){
    for(int i=0;i<kisiSayisi;i++){
        delete kisiDizi[i];
    }
}

int main(){
    Kisi **kisiler = new Kisi*[3];
    kisiler[0] = new Kisi("Ahmet");
    kisiler[1] = new Kisi("Mehmet");
    kisiler[2] = new Kisi("Ayse");
    Yazdir(kisiler, 3);
    Yoket(kisiler, 3);
    delete []kisiler;
    return 0;
}

```

## Özyineleme

```

#include <iostream>
using namespace std;
int Faktoriyel(int sayi){
    if(sayi <= 1) return 1;
    return sayi*Faktoriyel(sayi-1);
}
int Seri(int sayi){
    int sonuc=1;
    for(int i=2;i<=sayi;i++)sonuc*=i;
    return sonuc;
}
int main(){
    int sayi;
    cout<<"Sayi:";
    cin>>sayi;
    cout<<sayi<<"!="<<Faktoriyel(sayi)<<endl;
    return 0;
}

```

## Fibonacci

```

#include <iostream>
using namespace std;

```

```

int Fib(int n){
    if(n == 0) return 0;
    if(n == 1) return 1;
    return Fib(n-1)+Fib(n-2);
}
int FibSeri(int n){
    int once=0,simdi=1;
    if(n == 0) return 0;
    if(n == 1) return 1;
    for(int i=2;i<=n;i++){
        int simdi_yeni = once + simdi;
        once = simdi;
        simdi = simdi_yeni;
    }
    return simdi;
}
int main(){
    int n;
    cout<<"n:";
    cin>>n;
    cout<<"Fib("<n<<"):"<<Fib(n);
    return 0;
}

```

### İkili Arama

```

#include <iostream>
using namespace std;

bool ikiliArama(int sayilar[],int baslangic,int bitis,int aranan){
    int ortaindis = (baslangic+bitis) / 2;
    if(bitis < baslangic) return false;
    if(sayilar[ortaindis] == aranan) return true;
    if(aranan < sayilar[ortaindis])
        return ikiliArama(sayilar, baslangic, ortaindis-1,aranan);
    if(aranan > sayilar[ortaindis])
        return ikiliArama(sayilar, ortaindis+1, bitis,aranan);
}

int main(){
    int dizi[]={5, 26, 33, 47, 51, 58, 63, 69, 75, 88, 94, 101};
    int x;
    cout<<"Aranan:";
    cin>>x;
    if(ikiliArama(dizi,0,11,x)) cout<<"Sayi var";
    else cout<<"Sayi yok";
    return 0;
}

```

### Hanoi Kuleleri

```

#include <iostream>
using namespace std;

```

```
void Hanoi(int diskSayisi,char kaynak,char ara, char hedef){
    if(diskSayisi == 1)
        cout<<"Diski "<<kaynak<<" den "<<hedef<<" ye tasi"<<endl;
    else{
        Hanoi(diskSayisi-1,kaynak,hedef,ara);
        Hanoi(1,kaynak,ara,hedef);
        Hanoi(diskSayisi-1,ara,kaynak,hedef);
    }
}

int main(){
    int diskSayisi;
    cout<<"Hanoi Kuleleri"<<endl<<endl;
    cout<<"Disk Sayisi:";
    cin>>diskSayisi;
    Hanoi(diskSayisi,'A','B','C');
    return 0;
}
```