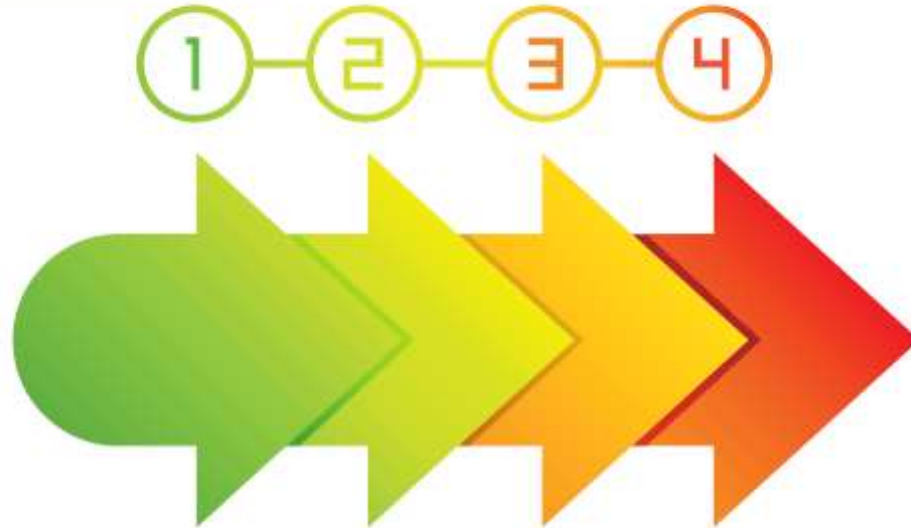


Programlama Yapıları

6. DOĐRUSAL MANTIK YAPISI İLE PROBLEM ÇÖZME



Bu bölümde;

- ✓ Bir probleme çözüm üretebilmek için doğrusal mantık yapısını kullanabilecek,
- ✓ Bir algoritma ve akış şeması oluştururken uygun yönergeleri kullanabilecek,
- ✓ Probleme çözüm üretirken yedi problem çözme aracını kullanabileceksiniz.

Doğrusal Mantık Yapısı

- Problem çözme sürecinde çok basit ve çok sık kullanılan yaklaşımlardan biri doğrusal mantık yapısıdır.
- Bilgisayara birbiri ardına algoritmanın başından sonuna kadar sırası ile işlemesi gereken komutları veren bir programcı, bu yaklaşımı kullanıyor demektir.

Algoritma ve Akış Şeması

Modül Adı

1. Yönerge

2. Yönerge

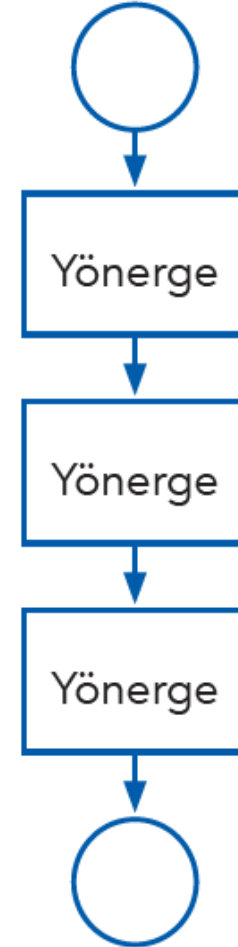
3. Yönerge

4.

....

....

XX. Bitir, Çıkış ya da Döndür(değer)

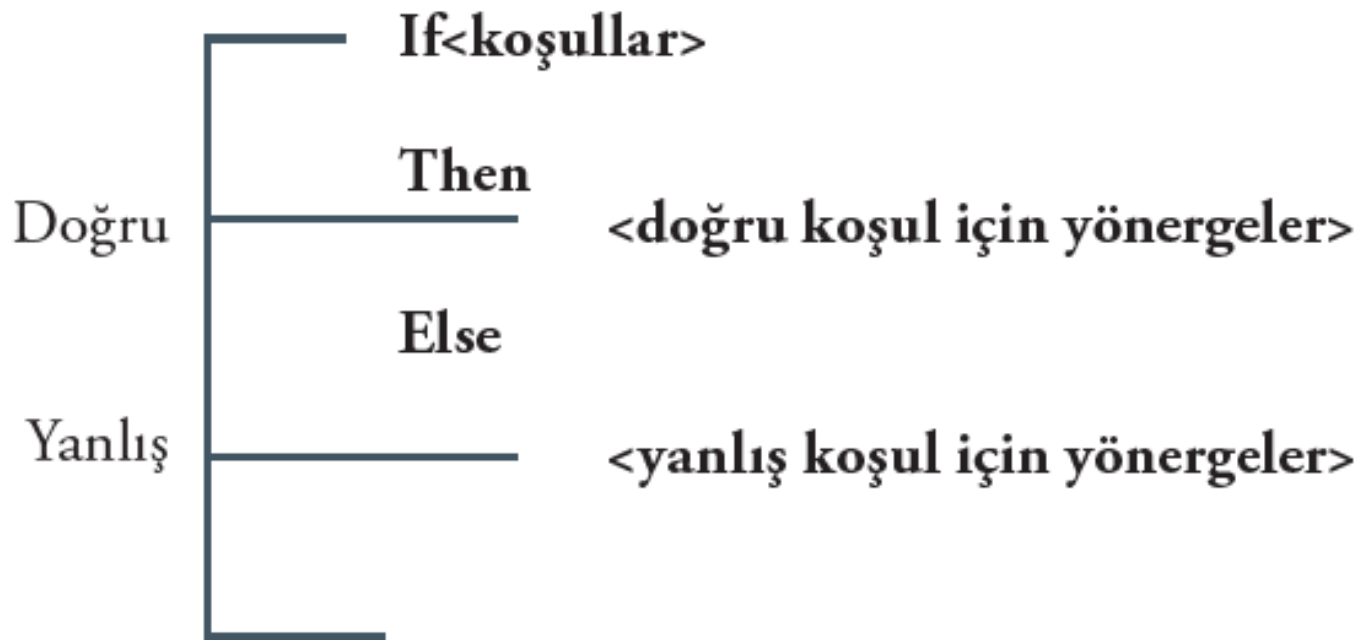


7. KARAR MANTIK YAPISI İLE PROBLEM ÇÖZME

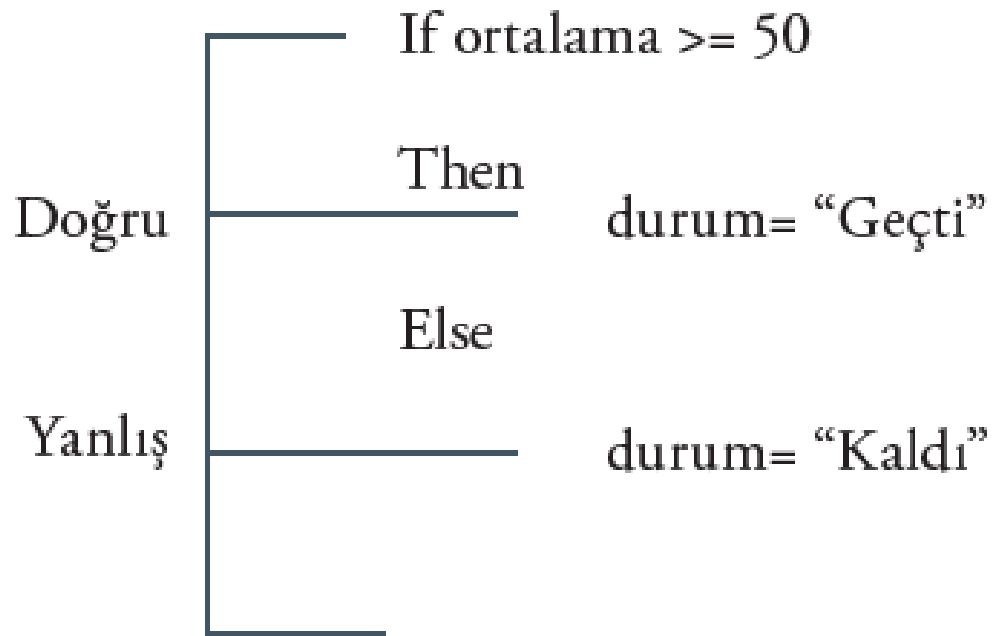


Karar Mantık Yapısı

- Karar yapıları, bilgisayara iki ya da daha fazla seçenek arasından seçim yapmak hakkı tanıyan önemli ve güçlü bir mantık yapısıdır.
- Eğer karar yapıları olmasaydı bilgisayarlar hızlı bir hesap makinesi olmanın ötesine gidemezdi.



Tek Koşullu Yapılar



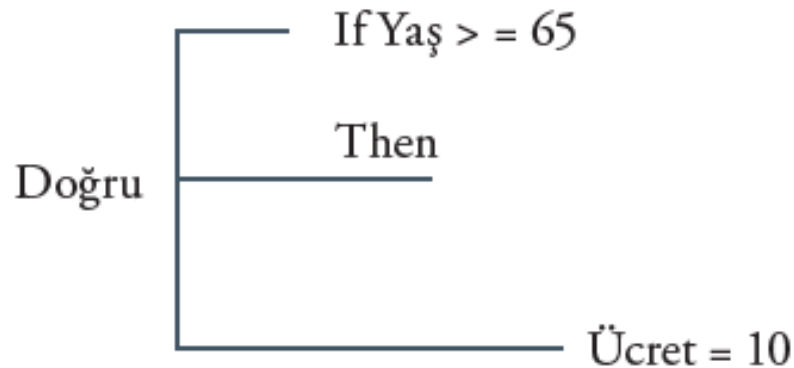
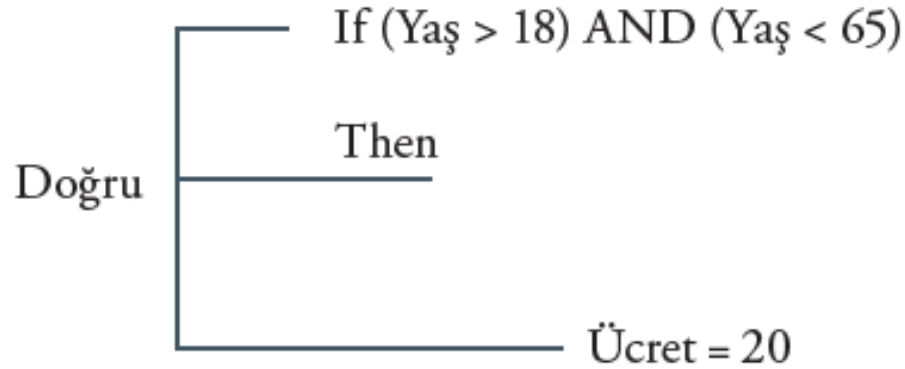
Çok Koşullu Karar Mantık Yapısı

- Düz Mantık
- Pozitif Mantık
- Negatif Mantık

Çok Koşullu Mantık Yapısı

Düz Mantık

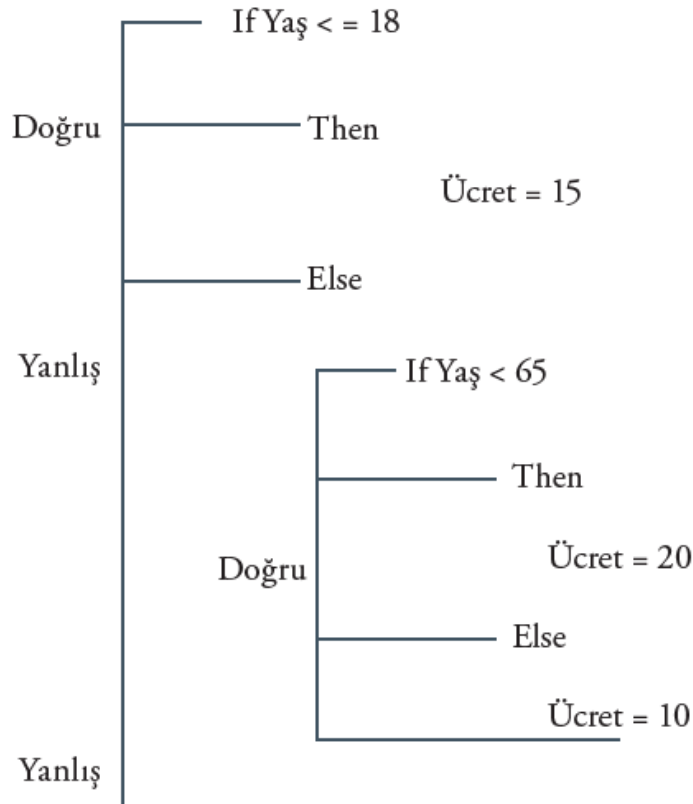
Düz mantık ile çalışan kararlarda bütün koşullar test edilir.



Çok Koşullu Mantık Yapısı

Pozitif Mantık Kullanımı

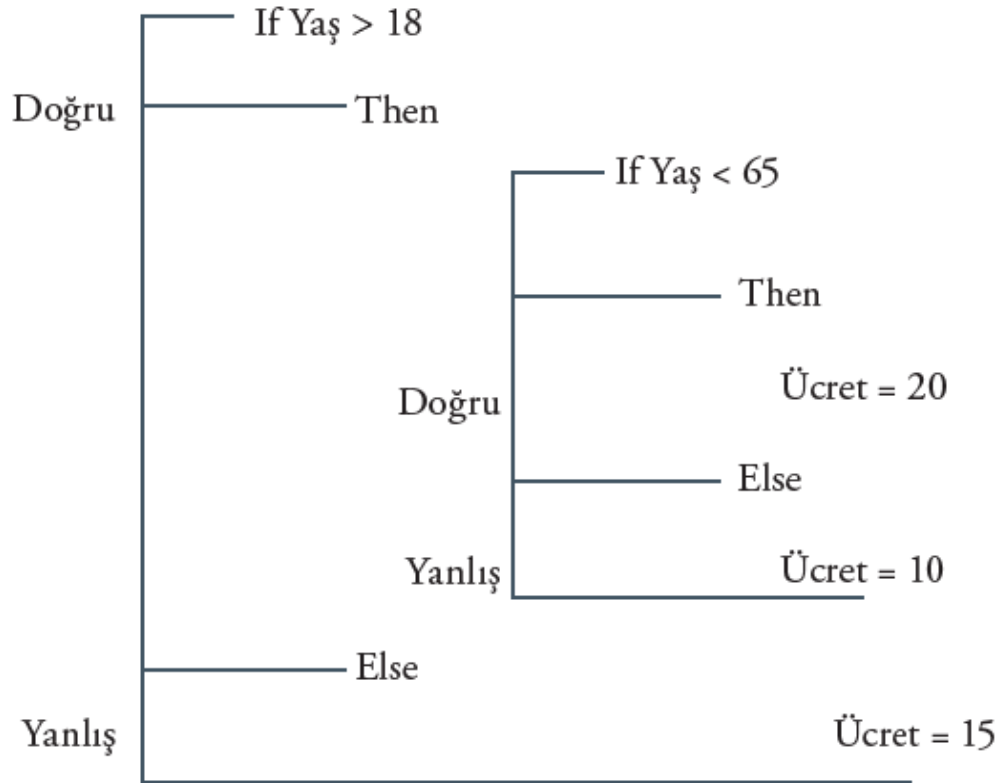
Bu yapı; kullanıldığında genellikle bilgisayardan, koşulun doğru olması durumunda işlem yapması, yanlış olması durumunda farklı bir karar vermesi beklenir. Böylece daha az adımda karar verilebilir.



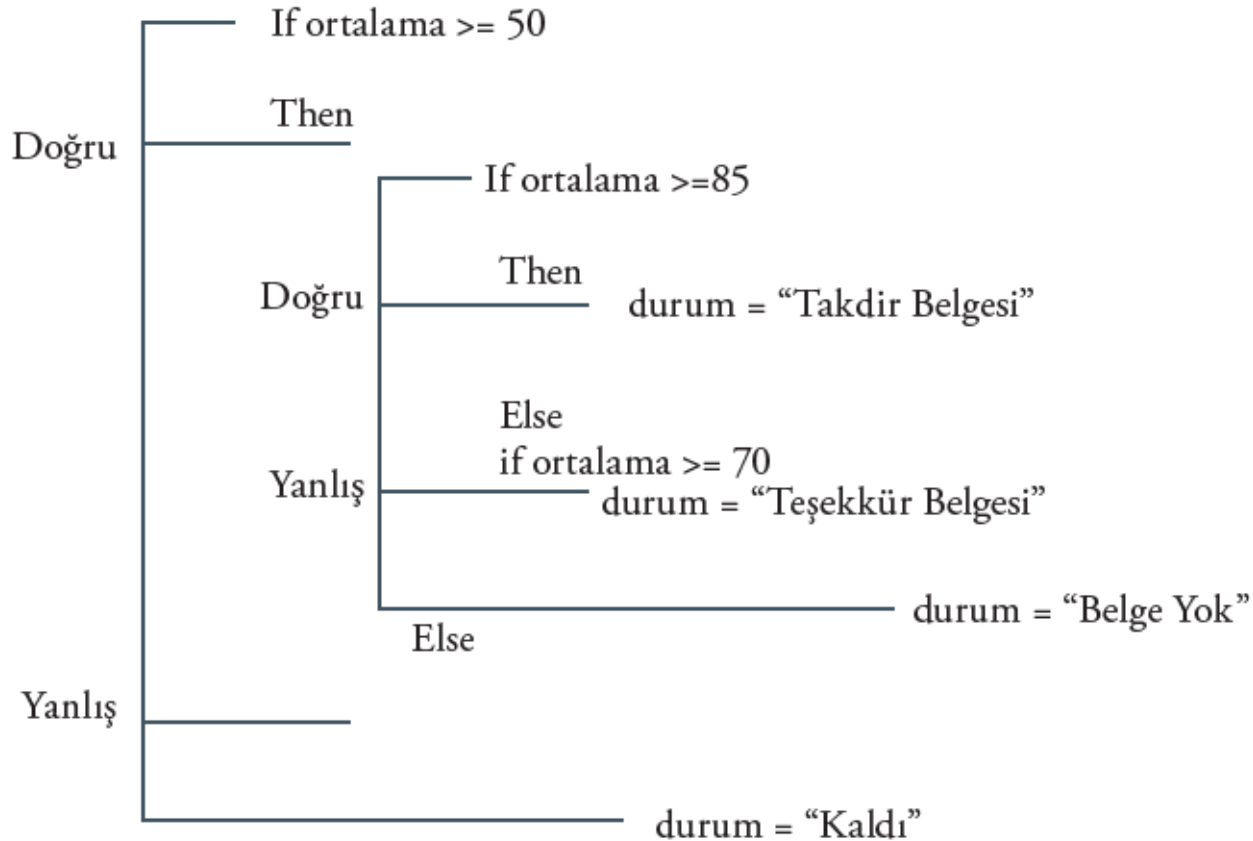
Çok Koşullu Mantık Yapısı

Negatif Mantık Kullanımı

Negatif mantık kullanıldığında bilgisayardan, koşulun doğru olması durumunda farklı yönergeleri takip etmesi beklenir.



İç İçe Karar Yapıları



8. DÖNGÜ YAPISI İLE PROBLEM ÇÖZME



Bu bölümde;

- ✓ Bir probleme çözüm üretebilmek için döngü yapısını kullanabilecek,
- ✓ Döngü yapısını karar yapıları ve doğrusal mantık ile birleştirebilecek,
- ✓ Problem çözme araçlarını kullanarak çözüm üretebilecek,
- ✓ Savaş ve birikeçleri problem çözümünde kullanabilecek,
- ✓ Problemin çözümü için iç içe döngüler oluşturabilecek,
- ✓ Döngü için kullanılacak 3 farklı türü ayırt edebilecek,
- ✓ Öz yinelemeyi kullanabileceksiniz.

Döngü Mantık Yapısı

- Bu yapı, tekrarlayan bir yapıdır.
- Çoğu problem, aynı işlemi farklı verileri kullanarak tekrar etmeyi gerektirir.

Döngü Mantık Yapısı

- **Arttırma**

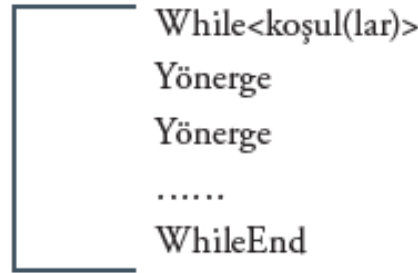
- Sayma ya da arttırma işlemi bir değişkene sürekli sabit bir değer ekleyerek gerçekleştirilir
- **sayaç = sayaç + 1**

- **Biriktirme**

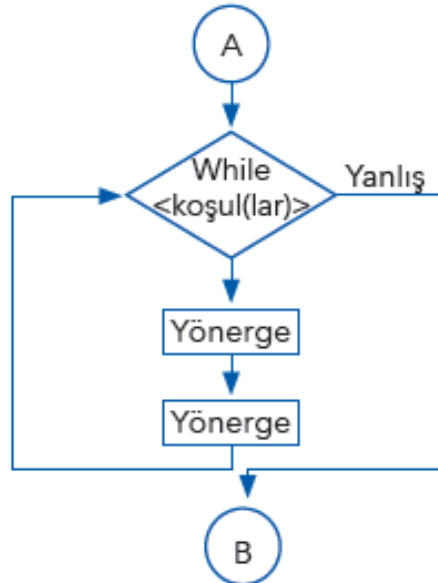
- Programın diğer bir görevi ise bir grup sayıyı toplayıp biriktirerek toplamı ya da sonucu bulmaktır.
- **toplam = toplam + değişken**

While/While End Döngüsü

Ele alacağımız ilk yapı, While/While End döngü yapısıdır. Bu döngü yapısı bilgisayara, koşul doğru olduğu sürece işlemleri tekrarlanmasını belirtir. Algoritma yapısı şu şekildedir:



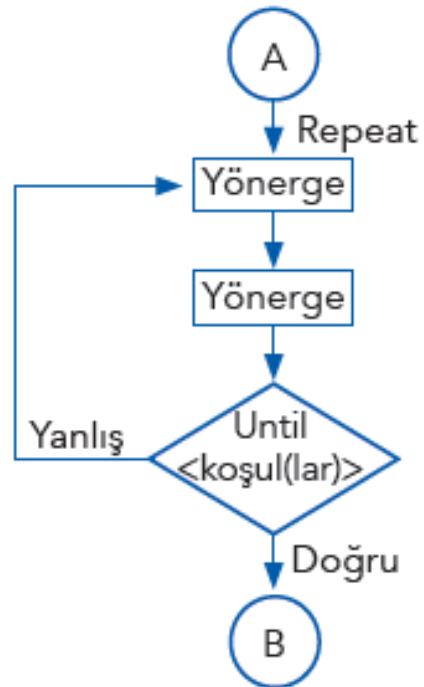
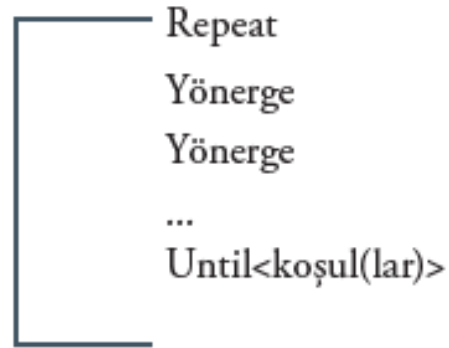
While/While End döngü yapısı tasarladığınız zaman, algoritmayı daha anlaşılır kılmak için girinti ve köşeli ayrıçlardan yararlanabilirsiniz. Bu yapıya ilişkin akış şeması ise şöyledir:



Repeat/Until Döngüsü

- While döngü yapısında döngü koşul doğru olduğu sürece çalışır, oysa Repeat yapısında döngü koşul doğru olduğunda durur,
- While döngüsünde koşul sürecin başında; Repeat yapısında ise döngünün sonunda kontrol edilir.

Repeat/Until Döngüsü



Otomatik Sayaç Döngüsü

- Bu yapıda döngünün her tekrarında bir değişkenin değeri arttırılır ya da azaltılır.

Otomatik sayaç döngüsüne ilişkin algoritma ve akış şeması şu şekildedir.

