

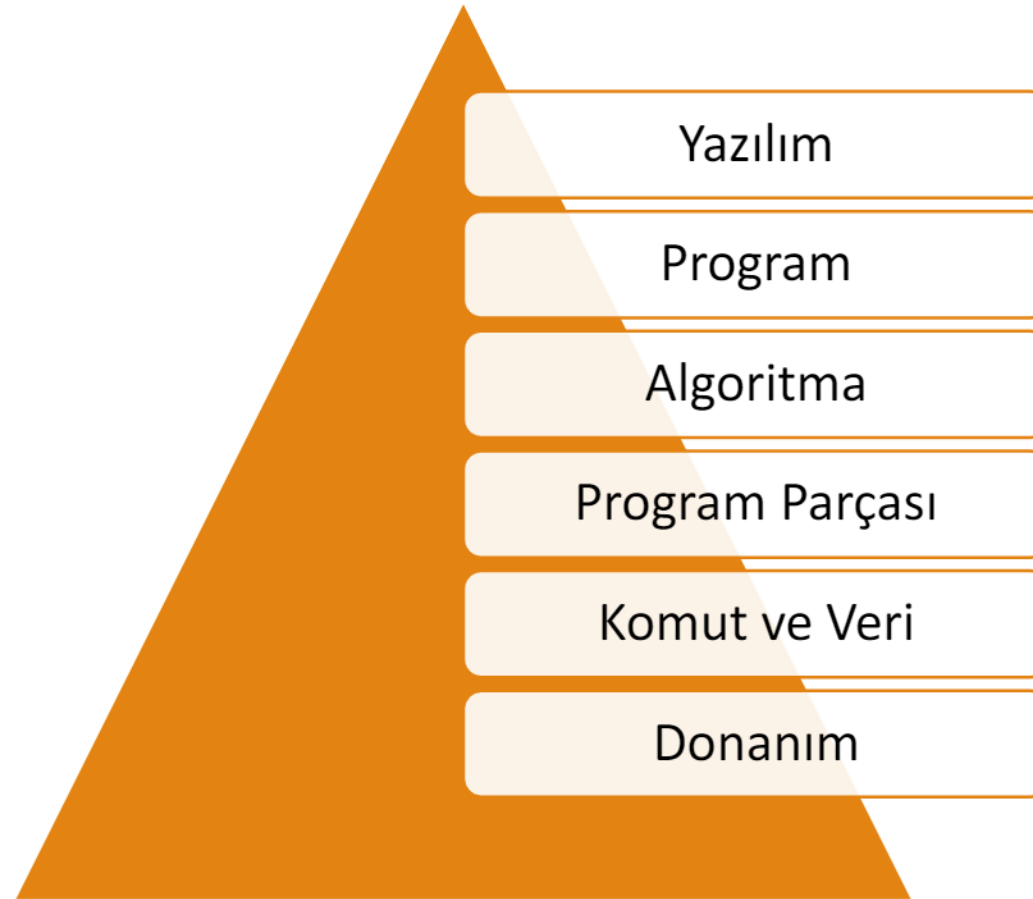
Veri Yapıları ve Algoritmalar

DR. ÖGR. ÜYESİ MEHMET AKİF BÜLBÜL

2023-2024 GÜZ YARIYILI

ALGORİTMİK PROGRAM TASARIMI VE AKIŞ ŞEMALARI

Program ve Yazılım Tasarımı



Program Tasarımı Yöntemleri

- Problem Analiz Şeması
- Yapısal Şema
- IPO Şeması (Input processing output chart)
- Akış Şemaları

Problem Analiz Şeması

Çözümü düzenlemek için önce programın beklentilerini analiz etmek gerekir. Bunun için en iyi yol, problemi dört aşamada ele almaktır:

1. Eldeki veri
2. Beklenen sonuç
3. Problemin çözüm süreci
4. Çözüm seçenekleri

Problem Analiz Şeması

| Eldeki Veri | Beklenen Sonuç |
|---|---|
| Problemde verilen ya da kullanıcı tarafından sağlanan veri | Sonuç için beklentiler, hangi bilginin nasıl biçimde sunulacağı |
| Problemın Çözüm Süreci | Çözüm Seçenekleri |
| İfade ve eşitlikler listesi, sıralama, arama, hesaplama vb. | Problemi çözebilmek için olası fikirler |

Bir örnek problem için problem analiz çizelgesinin nasıl olduğuna bir göz atalım: sınav ve performans puanlarına göre ortalama hesaplama ve geçme kalma durumunun kontrolü:

| Eldeki Veri | Beklenen Sonuç |
|---|---|
| 2 Yazılı ve 2 Performans Puanı | Geçme/Kalma Durumu |
| Problemın Çözüm Süreci | Çözüm Seçenekleri |
| <ul style="list-style-type: none">-Ortalama = $(\text{Yazılı 1} + \text{Yazılı 2} + \text{Performans 1} + \text{Performans 2})/4$-Geçme/Kalma Durumu= Eğer ortalama 50'den küçükse "Kaldı", değilse "Geçti" | Yazılı ve performans puanlarını girilecek değerler olarak tanımlama |

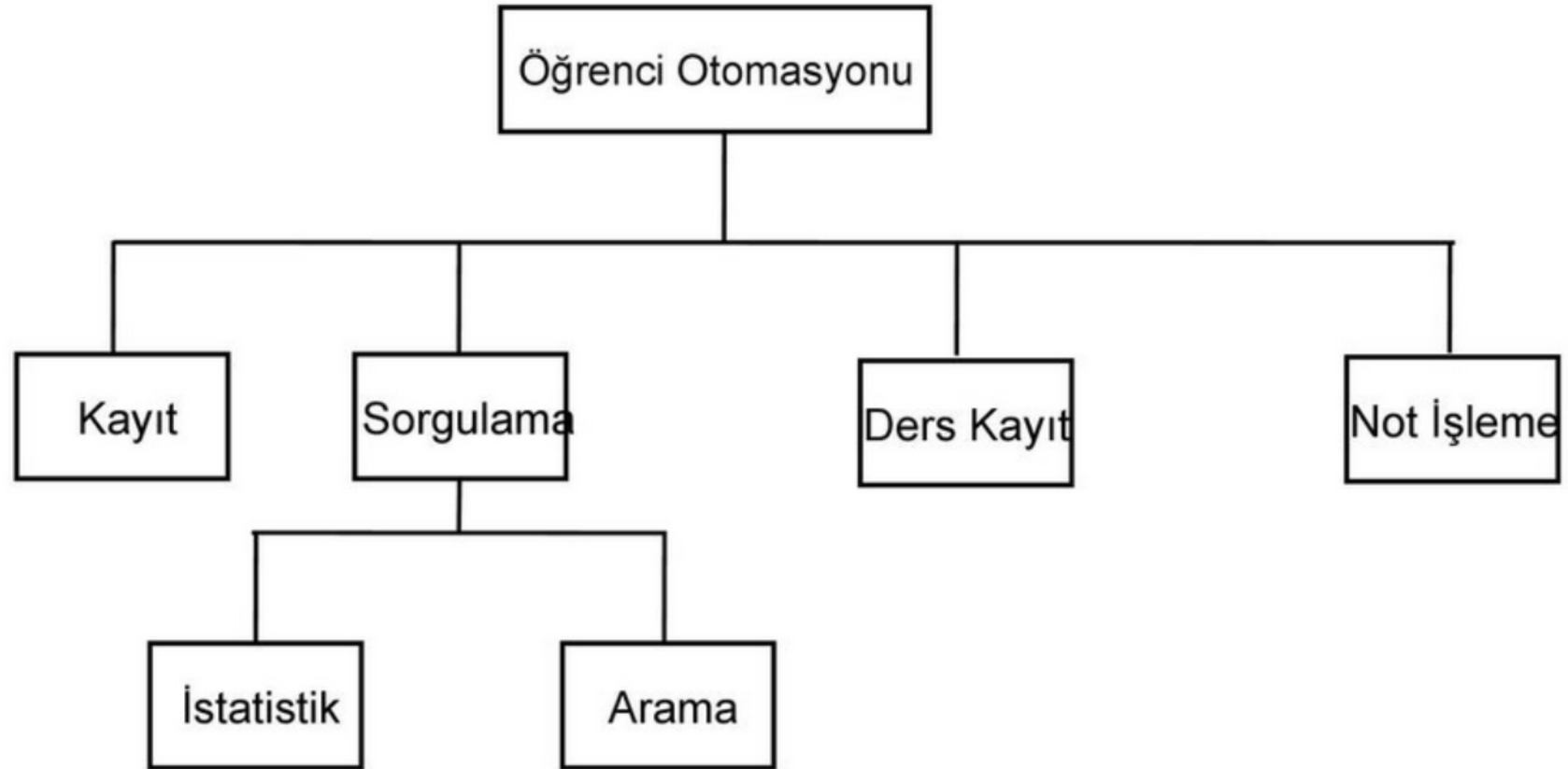
Yapısal Şema

Çözümə ulaşma yolunda, çözüm sürecini modüllere ayırmak ve süreçteki modüllerin birbiri ile etkileşimini görmek için modülleri birleştirmektir.

Yönetmelik etkileşim çizelgesi hazırlanırken yukarıdan aşağıya yaklaşım kullanılır.

Tüm modülleri kontrol eden bir ana kontrol mekanizması dâhilinde süreç yukarıdan aşağıya doğru işler.

Yapısal Şema



IPO Şeması

Dört bölümden oluşur

- Girdi,
- Süreç,
- Modül referansı ve
- Çıktı










IPO Şeması

| Girdi | Süreç | Modül Referansı | Çıktı |
|----------------------------------|---|---|---|
| Program için gerekli tüm veriler | Adım adım işlemler (Problem Analiz Çizelgesindeki 3 ve 4. Adımlar) | Etkileşim çizelgesindeki modüller | Tüm çıktı beklentileri (Problem Analiz Çizelgesindeki 1 ve 2. Adımlar) |

Geçme/Durumunu tespit etmek için GSC çizelgesi aşağıdaki gibidir.

| Girdi | Süreç | Modül Referansı | Çıktı |
|---------------------------------|--|-----------------|-------------|
| Sınav ve Performans Puanları | Sınav puanlarını gir. | Oku | Geçti/Kaldı |
| | Performans puanlarını gir. | Oku | |
| | Puan ortalamasını hesapla. | Hesapla | |
| | Puan ortalamasının 50'den küçük olup olmadığını kontrol et. | Karar | |
| | Geçme kalma durumunu ekrana yazdır. | Yazdır | |
| | Bitir. | Kontrol | |

Akış Şemaları

| Simge | İşlev |
|---|---------------------------------|
|  | Başla/Bitir |
|  | Giriş |
|  | Atama/İşlem |
|  | Denetim (Karar) |
|  | Çıkış |
|  | Döngü |
|  | Akış Yönü |
|  | Bağlaç |
|  | Önceden Tanımlı İşlem/Fonksiyon |

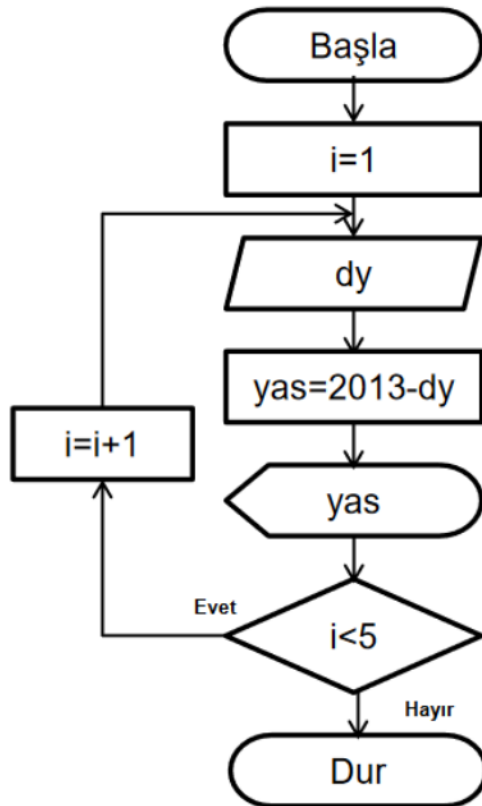
Akış Şemaları

AVANTAJLARI :

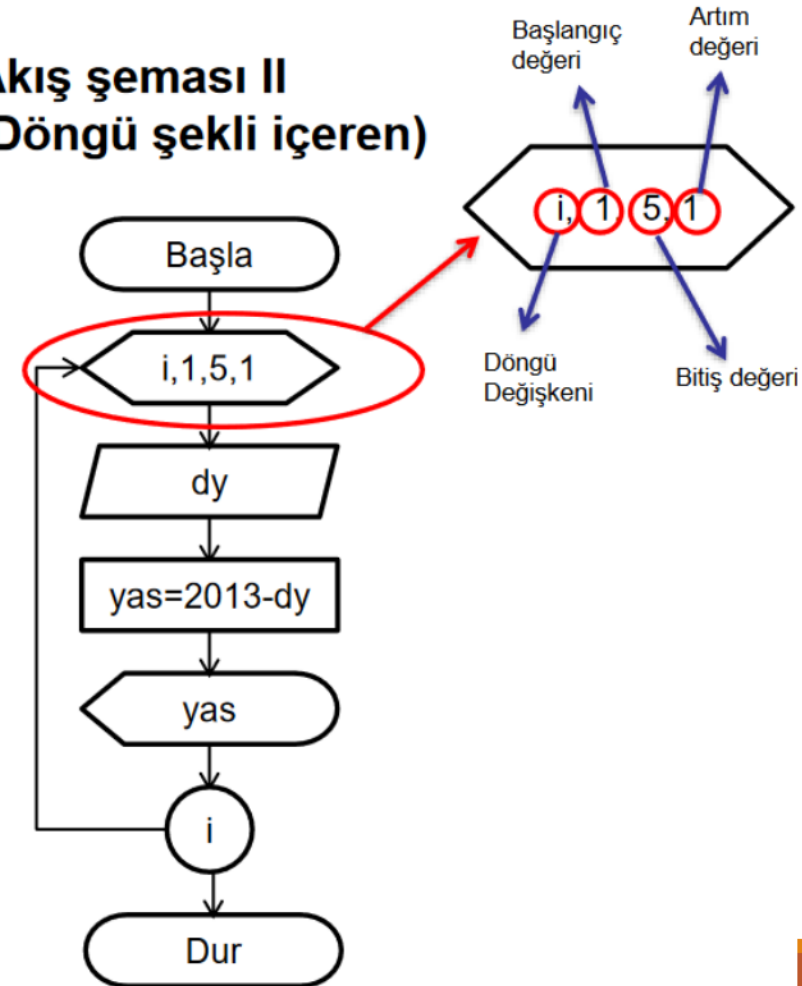
- 1) Birbiri ile ilgili adımlar arasındaki mantıksal ilişkiyi gösterirler.
- 2) İzlenmesi ve anlaşılması kolaydır.
- 3) Şartlara bağlı olarak ortaya çıkan olayların takip edilmesi mümkündür.
- 4) Belli bir standarda göre hazırlandıklarından herkesçe anlaşılabilir ve birden fazla kişi aynı şema üzerinde çalışabilir.

Akış Şemaları

Akış şeması I
(Döngü ifadesi içermeyen)

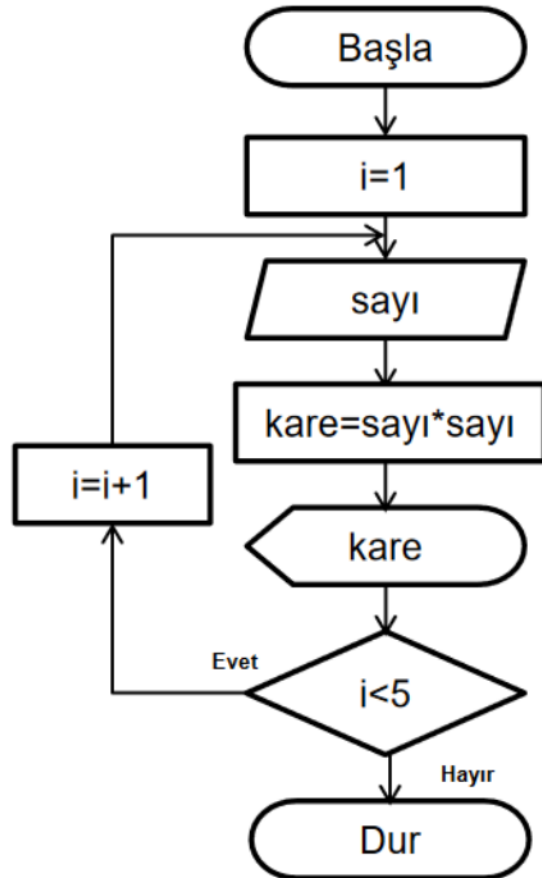


Akış şeması II
(Döngü şekli içeren)

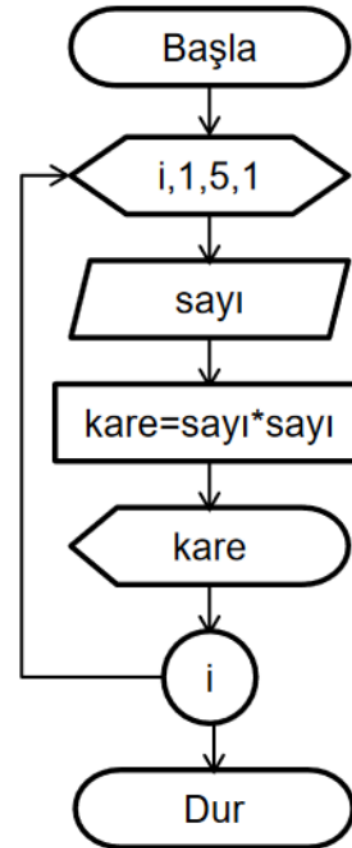


Akış Şemaları

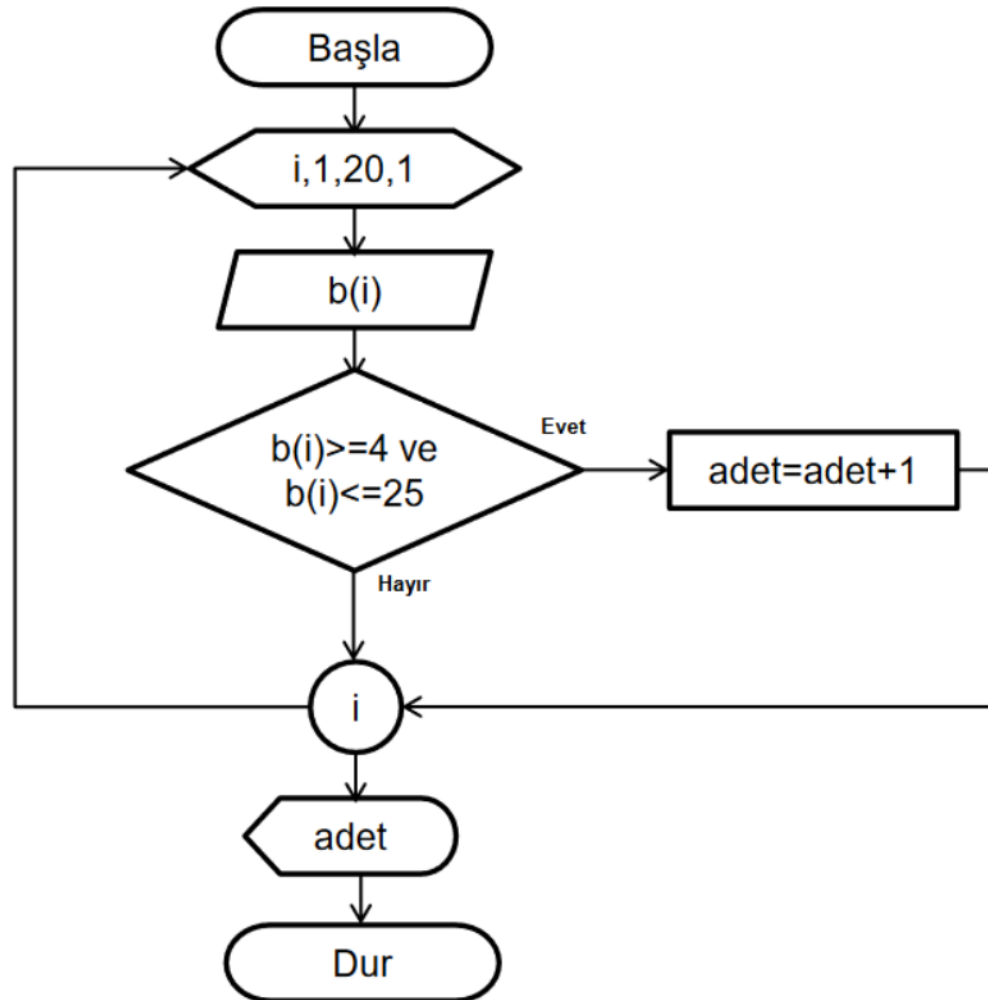
Kontrol İfadesiyle



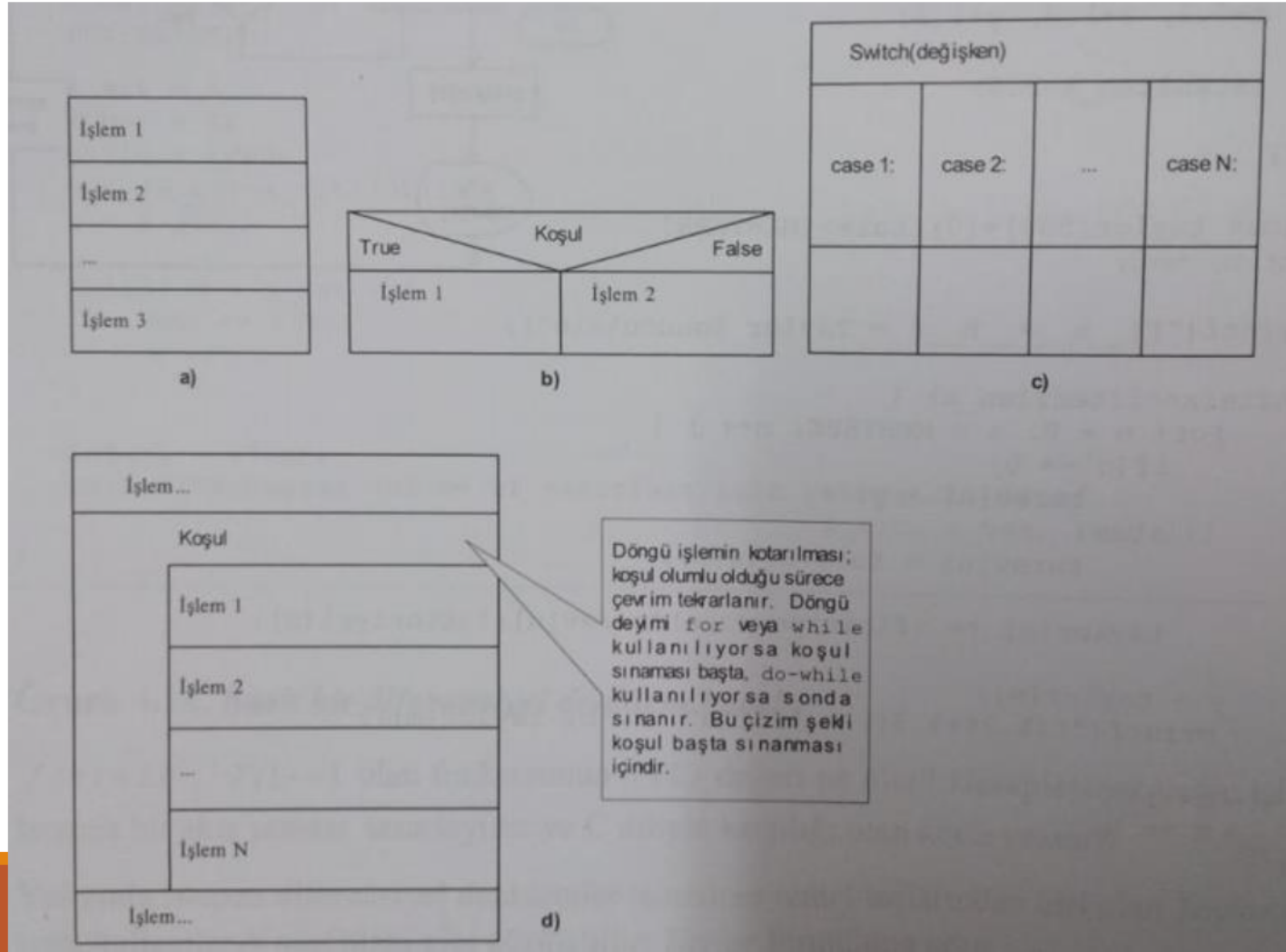
Döngü İfadesiyle



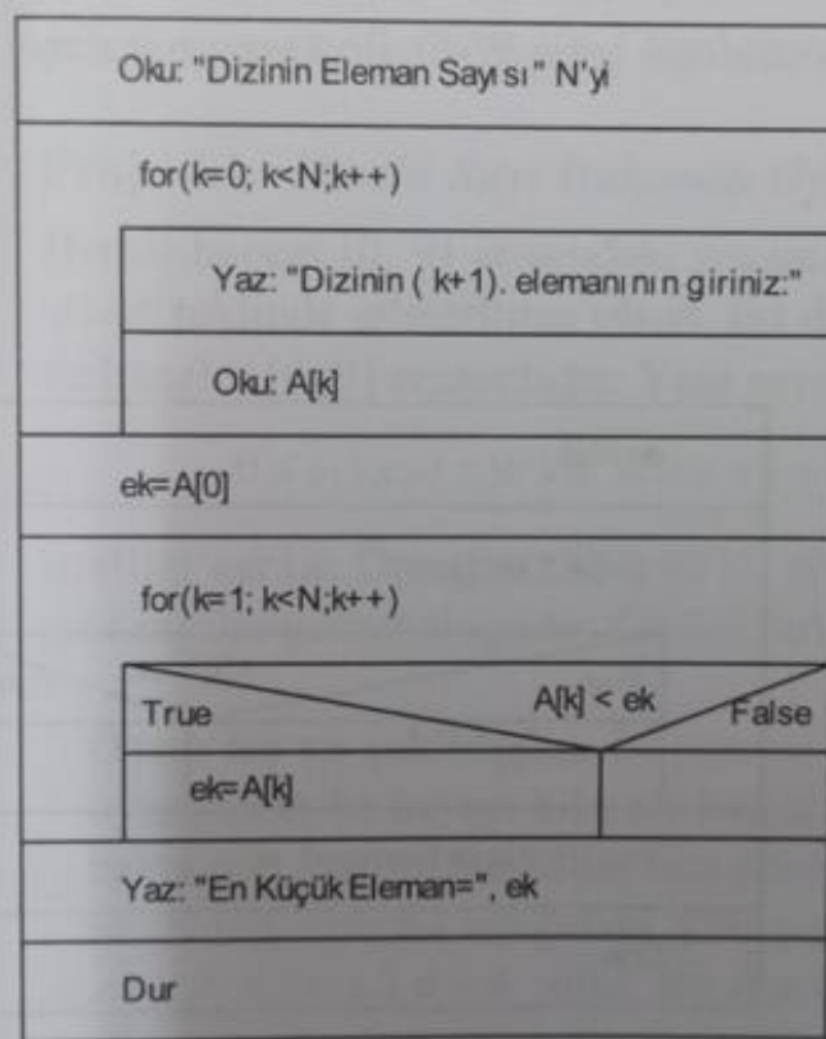
Akış Şemaları



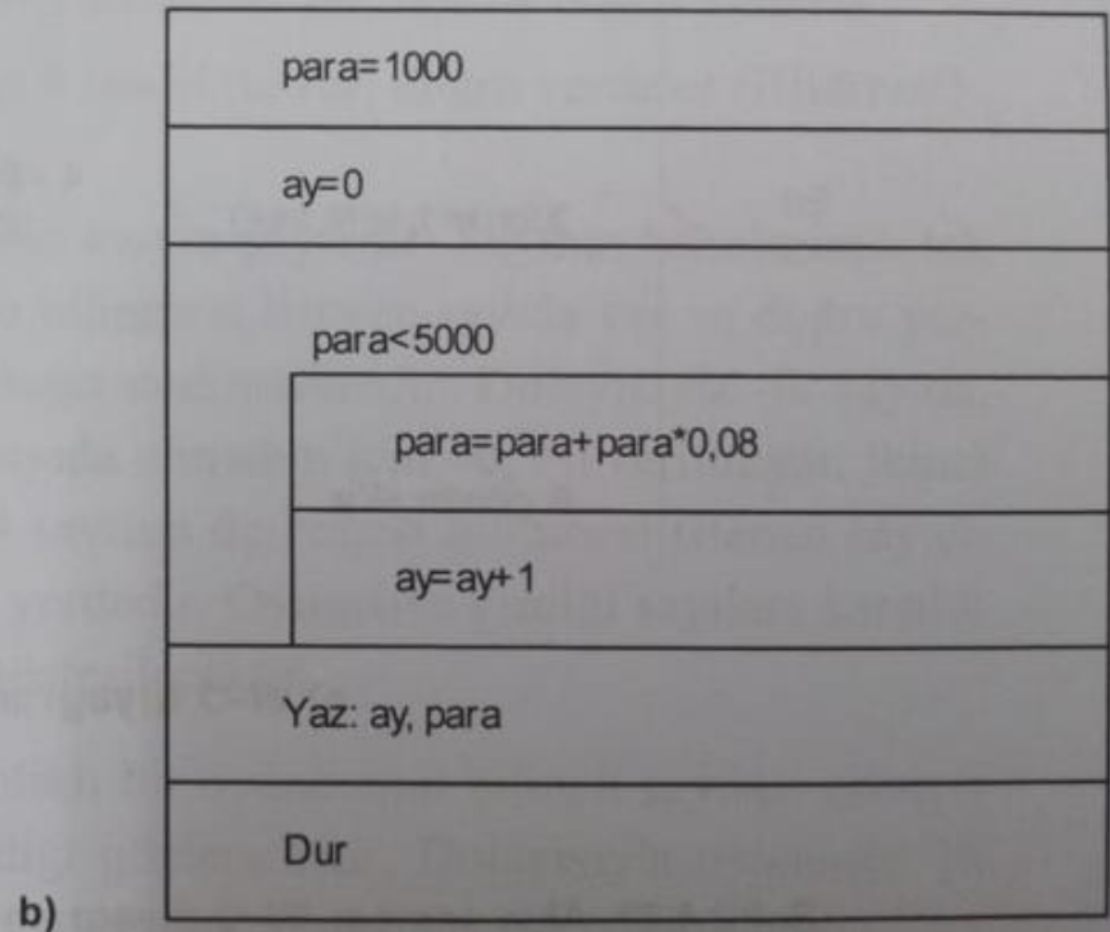
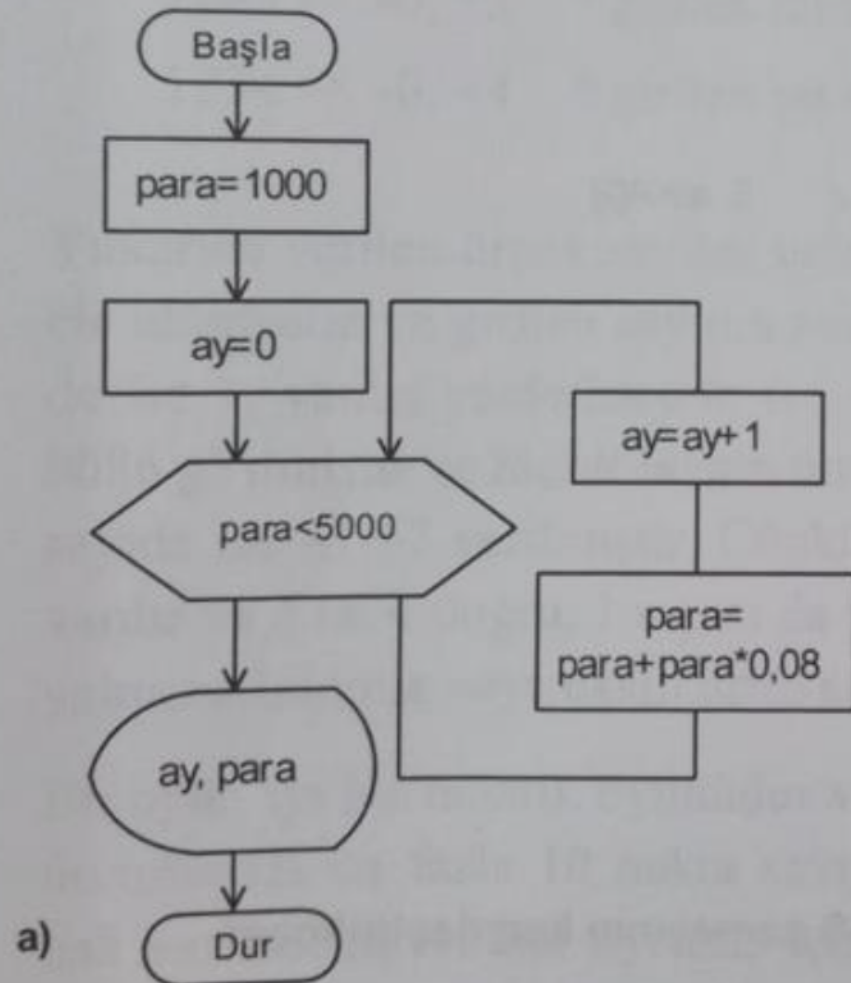
N-S (Nassi-Schneiderman) Şemaları



N-S (Nassi-Schneiderman) Şemaları



N-S (Nassi-Schneiderman) Şemaları



ÇÖZÜMÜN KODLANMASI

Akış şeması ve algoritmalar tamamlandıktan sonra istenilen bir programlama dili kullanılarak programın yazılması işlemine geçilir ki bu işleme “**programlama**” ya da “**kodlama**” adı verilir.

Kodlama sonucunda programın ne kadar hatasız çalıştığı, algoritmanın etkililiğine bağlıdır.