

BİL2001 Algoritmalar ve Veri Yapıları

ÖĞR. GÖR. DR. ALPER VAHAPLAR

2019 – 2020

BİL2001 – Algoritmalar ve Veri Yapıları

✓ Ders Saatleri

✓ 1. Şube

- Pazartesi günleri 5. 6. 7. ve 8. saatler
- 13:00 – 16:30
- Lab 13

✓ 2. Şube

- Çarşamba günleri 1. 2. 3. ve 4. saatler
- 08:30 – 12:00
- Lab 13

BİL2001 – Algoritmalar ve Veri Yapıları

✓ Materyal

- Ders Notları (Dersin sitesinde yayınlanacak)
- Kaynak Kitaplar
 - Çölkesen, R., *Veri Yapıları ve Algoritmalar Bilg. Prog. ve Yaz. Müh.*, Papatya Yayıncılık, 2002.
 - Carrano, F.M., *Data Abstraction and Problem Solving with C++: Walls and Mirrors* (4th Edition), Addison Wesley, 2004
 - Lafore, R., *Data Structures and Algorithms in Java* (2nd Edition), Sams, 2002.
- Internet (kaynak belirtmek kaydıyla)

BİL2001 – Algoritmalar ve Veri Yapıları

✓ Dersin Öğrenme Çıktıları:

- Temel ve tanımlamalı veri yapılarını bilme.
- Algoritmanın Çalışma Hızını ve Bellek Gereksinimini analiz edebilme.
- Temel Arama ve Sıralama algoritmalarını bilme.
- Liste, kuyruk, yığın ve ağaç veri modellerini bilme.

BİL2001 – Algoritmalar ve Veri Yapıları

Hafta	Konular
1	Giriş, Temel ve tanımlamalı veri yapıları
2	Veri Modelleri
3	Yığın ve Kuyruk Yapısı/Modeli
4	Bağlantılı Listeler ve Uygulamaları
5	Bağlantılı Listeler ve Uygulamaları (devamı)
6	Algoritmanın Zaman ve Bellek Gereksinimi
7	Ara sınav
8	Sıralama Algoritmaları
9	Sıralama Algoritmaları (devamı)
10	Arama Algoritmaları
11	Arama Algoritmaları (devamı)
12	Ağaç Veri Modeli
13	Ağaç Veri Modeli (devamı..)
14	Ağaç Uygulamaları

BİL2001 – Algoritmalar ve Veri Yapıları

✓ Değerlendirme

- Arasınav: %30
- Ödev/Quiz: %20
- Final: %50

BİL2001 – Algoritmalar ve Veri Yapıları

- ✓ Devamsızlık
 - %30 devamsızlık “izni” (hakkı değil...)
 - $14 \text{ imza} \times 0.3 = 4.2$
- ✓ 5 imzası eksik olan finale giremez...

BİL2001 – Algoritmalar ve Veri Yapıları

✓ Ders ile ilgili site:

- <http://csc.deu.edu.tr> – Moodle Sistemi

✓ İletişim mail adresi

- alper.vahaplar@deu.edu.tr

Algoritmalar ve Veri Yapıları

Giriş

Algoritma

- ✓ Algoritma Nedir?
- ✓ 9. yy – Ebu Abdullah Muhammed bin Musa el-Harezmi
- ✓ Bir problemi çözmeye yarayan, sonlu, iyi tanımlanmış, sıralı işlemler kümesi
- ✓ Bir problemin çözüm tarifi

Algoritma

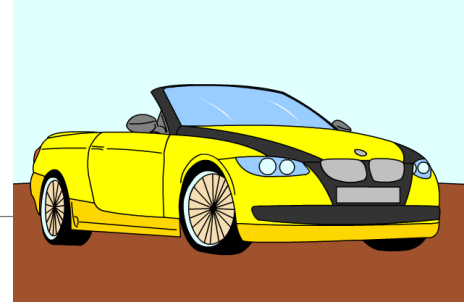
- ✓ Algorithm is
 - any well-defined computational procedure
 - that takes some value, or set of values, as input
 - and produces some value, or set of values, as output.
- ✓ Algorithm is
 - Logical and symbolic description of processes required for the solution of a problem.

Algoritma

✓ Algoritmanın Özellikleri

- Finiteness (Sonluluk)
- Definiteness (Kesinlik, açıklık)
- Valid Input / Output (Geçerli Giriş/Çıkış, Sonuç)
- Effectiveness (Etkinlik)
- Infallibility (Yanılmazlık, şaşmazlık)
- Performance (Performans)

Algoritma



- ✓ Problem: Parka halindeki bir aracı çalıştırıp hareket ettiriniz.
 1. Şoför kapısını aç
 2. Sürücü koltuğuna otur, kapıyı kapat
 3. Kemer tak
 4. Anahtarı kontak girişine yerleştir
 5. Koltuğu kendine göre ayarla
 6. Dikiz aynasını ve yan aynaları kendine göre ayarla
 7. Sol ayak ile en soldaki debriyaj pedalına sonuna kadar bas
 8. Vitesi boş konumuna getir
 9. Anahtarı, motor çalışıncaya kadar çevir, çalışınca bırak
 10. Vitesi 1 konumuna getir (Sol üstte)
 11. Sol sinyali çalıştır (Direksiyonun solundaki kolu aşağıya indir)
 12. Aynalardan trafiğin uygun olup olmadığını kontrol et
 13. Eğer uygun ise
 1. Eğer çekili ise el frenini indir
 2. Sağ ayak ile en sağdaki gaz pedalına hafifçe bas
 3. Sol ayağını debriyajdan hafifçe kaldır
 4. Hareket edince sol ayağını tamamen kaldır
 14. Değilse 12. adıma geri dön

Algoritma – Alıştırma 01

- ✓ Kağıttan uçak yapımını 7 yaşındaki bir çocuğun anlayacağı şekilde anlatan algoritmayı yazınız.

Algoritma

- ✓ İki sayının toplamını bulan algoritmayı yazınız.
 - Birinci sayıyı al
 - İkinci sayıyı al
 - İkisini topla
 - Sonucu söyle

Algoritma

✓ Girilen iki sayıdan büyük olanı gösteren algoritmayı yazınız.

1. A sayısını oku
2. B sayısını oku
3. Eğer A sayısı B sayısından büyükse
A sayısını yazdır
4. Değilse
B sayısını yazdır.
5. Bitir.

Algoritma

✓ Kullanıcı 0 girinceye kadar girilen sayıların **en küçüğünü** bulduran algoritmayı yazınız.

1. A sayısını al
2. Enküçük'e A sayısını ata ($\text{Enküçük} \leftarrow A$)
3. A sayısı 0'dan farklı olduğu sürece
 - a. A sayısını al
 - b. Eğer $A \text{ sayısı} < \text{Enküçük}$ ise $\text{Enküçük} \leftarrow A$
4. Enküçük'ü yazdır.
5. Bitir.

DOĞRU MU???

Algoritma

✓ Kullanıcı 0 girinceye kadar girilen sayıların **en küçüğünü** bulduran algoritmayı yazınız.

1. A sayısını al
2. Enküçük'e A sayısını ata ($\text{Enküçük} \leftarrow A$)
3. A sayısı 0'dan farklı olduğu sürece
 - a. Eğer $A \text{ sayısı} < \text{Enküçük}$ ise $\text{Enküçük} \leftarrow A$
 - b. A sayısını al
4. Enküçük'ü yazdır.
5. Bitir.

Algoritma

✓ Kullanıcı 0 girinceye kadar girilen sayıların **toplamını** bulduran algoritmayı yazınız.

1. A sayısını al
2. Toplam $\leftarrow 0$
3. A sayısı 0'dan farklı olduğu sürece
 - a. Toplam' a A sayısını ekle
 - b. A sayısını al
4. Toplam'ı yazdır.
5. Bitir.

Akış Şemaları – Flowcharts

- ✓ Algoritmanın adımlarını görsel olarak ifade edebilmek için kullanılan semboller topluluğudur.
- ✓ Farklı geometrik şekillerle gösterilir.
- ✓ Küçük problemler için kullanışlıdır.

Akış Şemaları – Flowcharts



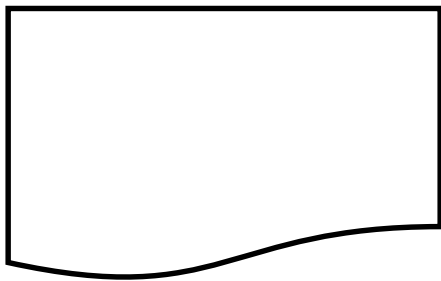
- Terminator
(Sonlandırıcı)

- Başlangıç veya bitişi ifade eder.



- Manuel Input
(Veri Girişi)

- Kullanıcı tarafından veri girileceğini gösterir.



- (Output, Document)
(Çıkış)

- Kullanıcıya veya ekrana mesaj verileceğini belirtir.



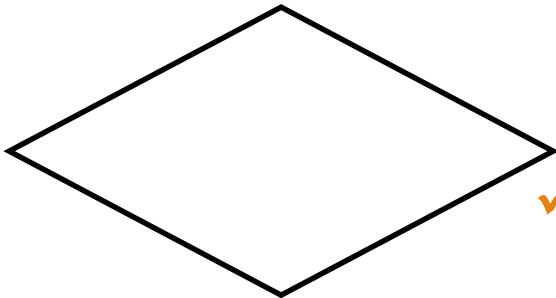
- (Output, screen)
(Çıkış)

Akış Şemaları – Flowcharts



✓ Process (İşlem)

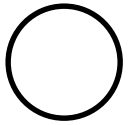
- Hesaplama yapıldığı veya veri işleme sürecini belirtir.



✓ Decision (Karar)

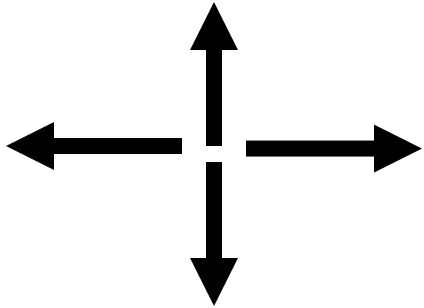
- Bir karşılaştırma, koşul ve sonucunda karar verileceğini ifade eder.

Akış Şemaları – Flowcharts



✓ Connector
(Bağlayıcı)

- Bir akış şemasının farklı kısımlarının bağlanması için kullanılır.



✓ Arrows, Directions
(Oklar)

- Algoritmanın akış yönünü gösterir.

Akış Şemaları – Örnek

Örnek : Gerekli parametreleri kullanıcı tarafından girilen bir silindirin hacmini hesaplayıp yazdırın.

Algoritma :

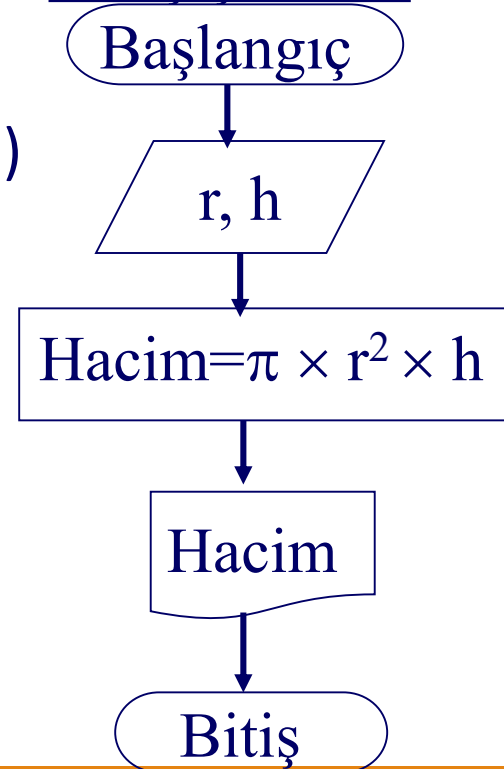
1. Kullanıcı yarıçapı (r) ve yüksekliği (h) girsin.

2. Silindirin hacmini bulun.

$$\text{Hacim} = \pi \times r^2 \times h$$

3. Silindirin hacmini yazdırın.

Akış şeması :

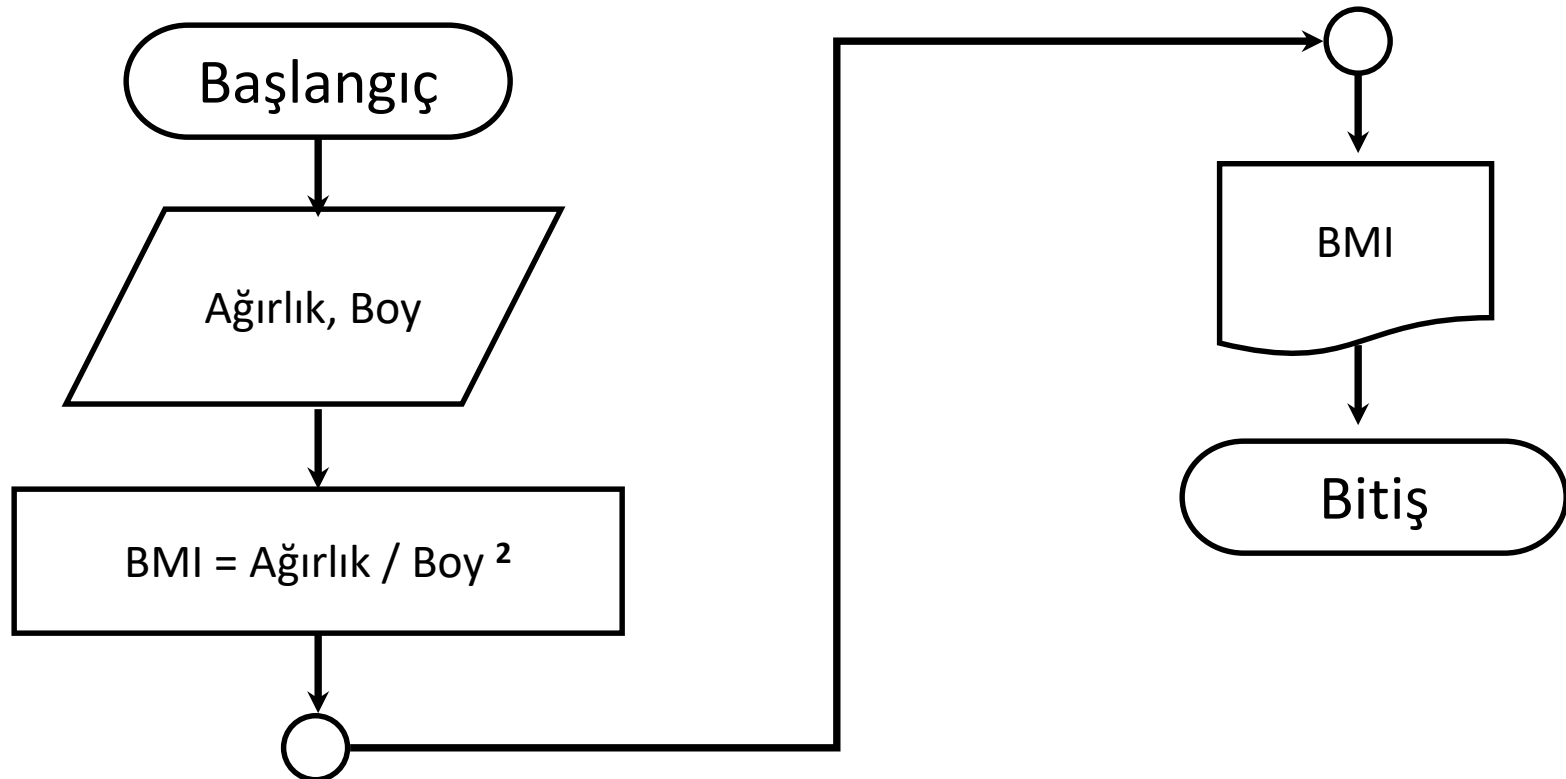


Akış Şemaları – Örnek

- ✓ Verilecek değerler için kişinin Vücut Kitle Endeksini (Body-Mass Index – BMI) hesaplayan algoritmayı yazınız ve akış şemasını çiziniz.
- ✓ $BMI = \text{Ağırlık (kg)} / [\text{Uzunluk (m)}]^2$

1. Başla
2. Ağırlık, Boy al
3. $BMI = \text{Ağırlık} / (\text{boy} * \text{boy})$
4. BMI yazdır
5. Bitir

Akış Şemaları – Örnek



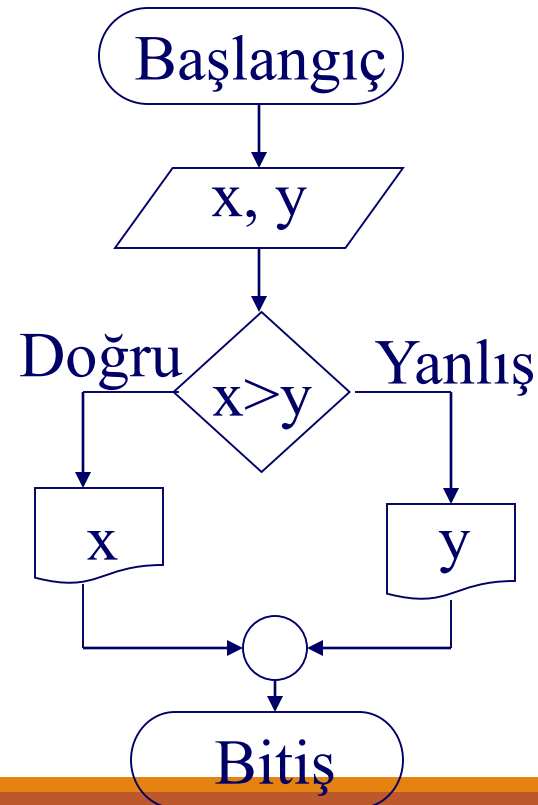
Akış Şemaları – Örnek

Örnek: Kullanıcının gireceği iki sayıdan büyük olanı yazdırın.

Algoritma :

1. Kullanıcı iki sayı girsin.
2. İki sayıdan büyük olanı bulun.
3. Bulunan sayıyı ekrana yazdırın.

Akış Şeması :



Akış Şemaları – Örnek

- ✓ Örnek: Katsayıları verilen ikinci dereceden bir denklemin reel hesaplayan algoritmayı yazınız.

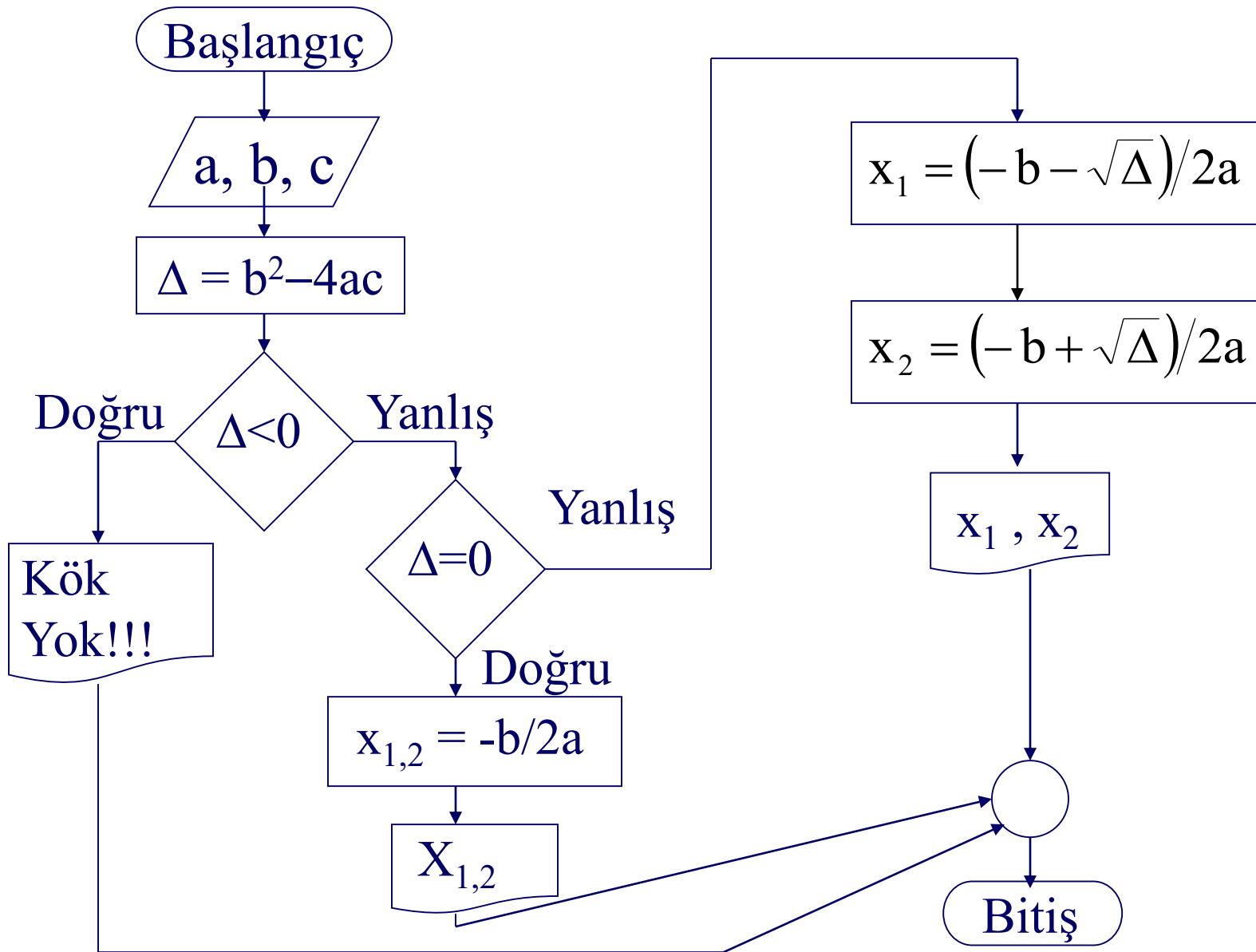
$$ax^2 + bx + c = 0$$

1. a, b ve c değerlerini al
2. Δ değerini hesapla ($\Delta = b^2 - 4ac$)
3. Eğer $\Delta < 0$ ise “Reel kök yok” mesajı ver
4. Eğer $\Delta = 0$ ise $x_{1,2} = -b/2a$
5. Eğer $\Delta > 0$ ise

6. Bitir.

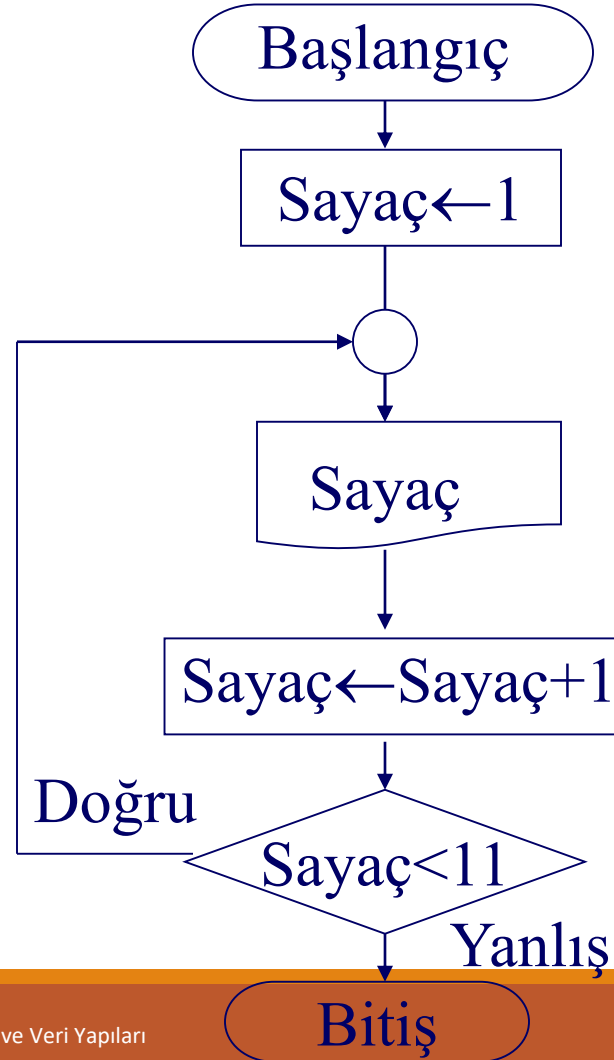
$$x_1 = \frac{-b - \sqrt{\Delta}}{2a}$$

$$x_2 = \frac{-b + \sqrt{\Delta}}{2a}$$



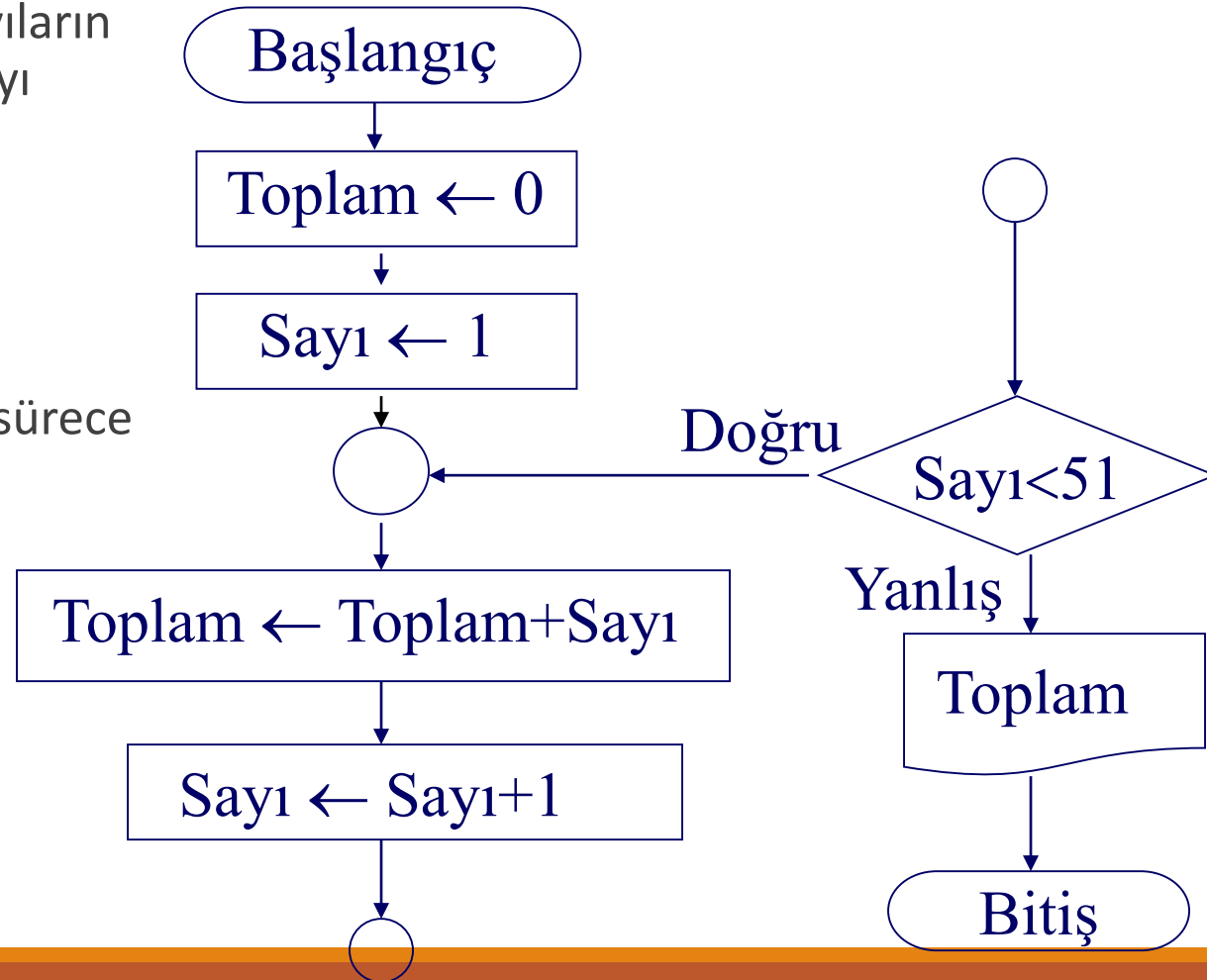
Akış Şemaları – Örnek

- ✓ 1'den 10'a kadar sayıları yazdıran algoritma
- 1. $\text{Sayaç} \leftarrow 1$
- 2. $\text{Sayaç}'ı$ yaz.
- 3. $\text{Sayaç} \leftarrow \text{Sayaç} + 1$
- 4. Sayaç 11'den küçük ise 2. adıma git.
- 5. Bitir.



Akış Şemaları – Örnek

- ✓ 1'den 50'ye kadar olan sayıların toplamını bulan algoritmayı yazınız.
1. $\text{Toplam} \leftarrow 0$
 2. $\text{Sayı} \leftarrow 1$
 3. Sayı 51'den küçük olduğu sürece
 - i. $\text{Toplam} \leftarrow \text{Toplam} + \text{Sayı}$
 - ii. $\text{Sayı} \leftarrow \text{Sayı} + 1$
 4. Toplam'ı yazdır
 5. Bitir.



Akış Şemaları – Örnek

$$\frac{\sum_{i=1}^N \mathbf{x}_i}{N} = \frac{\mathbf{x}_1 + \mathbf{x}_2 + \dots + \mathbf{x}_N}{N} = ?$$

- Kullanıcının gireceği N adet verinin ortalamasını bulup ekrana yazdırın. Veri sayısını belirleyen N, kullanıcı tarafından girilecektir.

Akış Şemaları – Örnek

1. Kullanıcı N değerini girsin.
2. Toplam $\leftarrow 0$
3. Sayaç $\leftarrow 0$
4. Sayaç $< N$ olduğu sürece
 - 4.1. Kullanıcı bir Sayı girsin.
 - 4.2. Toplam \leftarrow Toplam + Sayı
 - 4.3. Sayaç \leftarrow Sayaç + 1
5. Ortalama \leftarrow Toplam / N
6. Ortalamayı ekrana yazdır.
7. Bitir.

Akış Şemaları – Örnek

