

# Algoritma ve Akış Diagramları

## ALGORİTMANIN HAZIRLANMASI

Algoritma, herhangi bir sorunun çözümü için izlenecek yol anlamına gelmektedir. Çözüm için yapılması gereken işlemler hiçbir alternatif yoruma izin vermeksizin sözel olarak ifade edilir. Diğer bir deyişle algoritma verilerin bilgisayara hangi çevre biriminden girileceğinin, problemin nasıl çözüleceğinin, hangi basamaklardan geçirilerek sonuç alınacağıının, sonucun nasıl ve nereye yazılacağıının sözel olarak ifade edilmesi biçiminde tanımlanabilir.

Algoritma hazırlanırken, çözüm için yapılması gerekli işlemler, öncelik sıraları gözönünde bulundurularak ayrıntılı bir biçimde tanımlanmalıdır. Aşağıda algoritma hazırlanmasına ilişkin örnekler yer almaktadır.

ÖRNEK 1: Verilen iki sayının toplamının bulunmasının algoritması aşağıdaki gibi yazılır:

### Algoritma

#### Adım 1-Başla

#### Adım 2-Birinci sayıyı oku

#### Adım 3-İkinci sayıyı oku

#### Adım 4-İki sayıyı topla

#### Adım 5-Dur

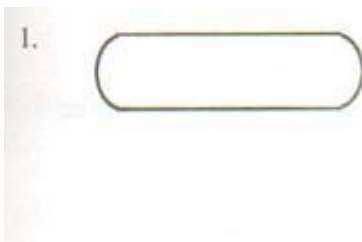
Algoritmaya dikkat edilirse işlemlerin sıralanmasında, işlem önceliklerinin gözönünde bulundurulduğu görülür. Ayrıca algoritma yazımı sorun çözümünün başladığını gösteren "BAŞLA" ifadesi ile başlamakta ve işlemlerin bittiğini belirten "DUR" ifadesi ile sona ermektedir.

## AKIŞ ŞEMALARI(DİYAGRAMLARI)

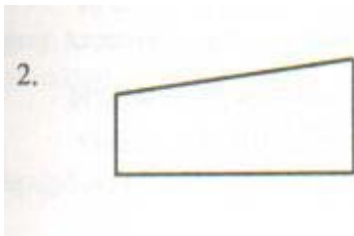
Herhangi bir sorunun çözümü için izlenmesi gerekli olan aritmetik ve mantıksal adımların söz veya yazı ile anlatıldığı algoritmanın, görsel olarak simge ya da sembollerle ifade edilmiş şekline "akış şemaları" veya FLOWCHART adı verilir. Akış şemalarının algoritmadan farkı, adımların simgeler şeklinde kutular içine yazılmış olması ve adımlar arasındaki ilişkilerin ve yönünün oklar ile gösterilmesidir.

Programın saklanacak esas belgeleri olan akış şemalarının hazırlanmasına, sorun çözümlenmesi sürecinin daha kolay anlaşılır biçime getirilmesi, iş akışının kontrol edilmesi ve programın kodlanmasının kolaylaştırılması gibi nedenlerle başvurulur. Uygulamada çoğunlukla, yazılacak programlar için önce programın ana adımlarını (bölümlerini) gösteren genel bir bakış akış şeması hazırlanır. Daha sonra her adım için ayrıntılı akış şemalarının çizimi yapılır.

Akış şemalarının hazırlanmasında aşağıda yer alan simgeler kullanılır.



Algoritmanın başladığını ya da sona erdiğini belirtmek için kullanılır.



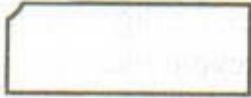
Klavye aracılığı ile giriş ya da okuma yapılacağını gösterir.

Yazıcı (printf) aracılığı ile çıkış yapılacağını gösterir.

3.

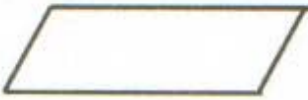


4.



Kart okuyucu aracılığıyla giriş yapılacağını gösterir. (Önemsiz)

5.



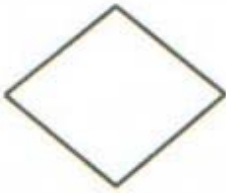
Araç belirtmeden giriş ya da çıkış (scanf) yapılacağını gösterir.

6.



Hesaplama ya da değerlerin değişkenlere aktarımını gösterir.

7.



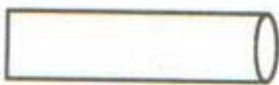
Aritmetik ve mantıksal ifadeler için karar verme ya da karşılaştırma durumunu gösterir.

8.



Diskten okuma ya da diskete yazmayı gösterir. (Önemsiz)

9.



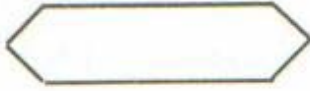
Disketten okuma ya da diskete yazmayı gösterir.(Önemsiz)

10.



Teyp kütüğünü gösterir.(Önemsiz)

11.



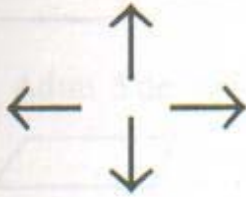
Yapılacak işler birden fazla sayıda yinelenecek ise diğer bir deyişle iş akışında çevrim(döngü) var ise bu sembol kullanılır.

12.



Akış diyagramında iki nokta arası ilişkiyi gösterir.Döngü sonunu göstermek için ya da diyagramın çizilemediği durumlarda kullanılır. Burada i herhangi bir sembol olabilir. (Önemsiz)

13.



Oklar işin akış yönünü gösterir.

Akış şemaları içerik ve biçimlerine göre genel olarak üç grupta sınıflandırılabilirler.

### DOĞRUSAL AKIŞ ŞEMALARI

İş akışları,giriş,hesaplama,çıkış biçiminde olan akış şemaları bu grup kapsamına girer.

ÖRNEK 2:İki sayının çarpımının bulunmasıyla ilgili algoritma şöyledir.

#### Değişkenler

**A:**Birinci sayıyı,

**B:**İkinci sayıyı,

**C:**İki sayının çarpımını (A\*B)gösterecek.

#### Algoritma

Adım 1-Başla

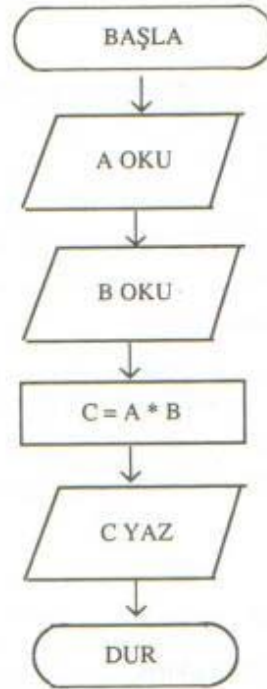
Adım 2-A'yı oku

Adım 3-B'yi oku

Adım 4-C=A\*B yi hesapla

Adım 5-C'yi yaz

Adım 6-Dur

**Akış Şeması**

Algoritma adımlarında kullanılması gereken semboller yukarıdaki şekilde görülmektedir.Örneğin "DUR" ve "BAŞLA" işlemleri için aynı sembol kullanılmaktadır.Adım 2 ve 3 için kullanılan sembol,bilgisayara değerlerin dışarıdan girildiğini gösterir.Oklar ise işin akış yönünü gösterir. Adım 4'te kullanılan sembol  $C=A*B$  gibi aritmetik işlemler için kullanılır.

Örnek 3:İki sayının farkını ve bölümünü bulup yazıcı ile yazan algoritma ve akış şeması şöyledir.

**Değişkenler**

**A:**Birinci sayı

**B:**İkinci sayı

**D:**İki sayının farkını  $(A-B)$

**E:**İki sayının bölümünü  $(A/B)$

**Algoritma**

Adım 1-Başla

Adım 2-A'yı oku

Adım 3-B'yi oku

Adım 4- $D=A-B$

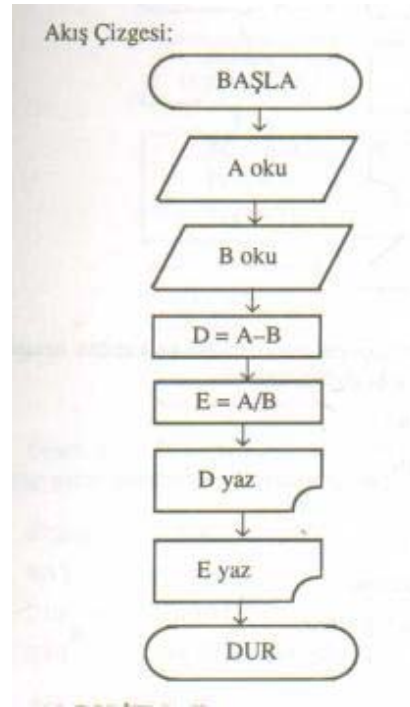
Adım 5- $E=A/B$

Adım 6-D'yi yaz

Adım 7-E'yi yaz

Adım 8-Dur

**Akış Şeması**



ÖRNEK 4:İki sayının toplamlarının karesini ve küpünü hesaplayıp yazan akış şeması şöyledir.

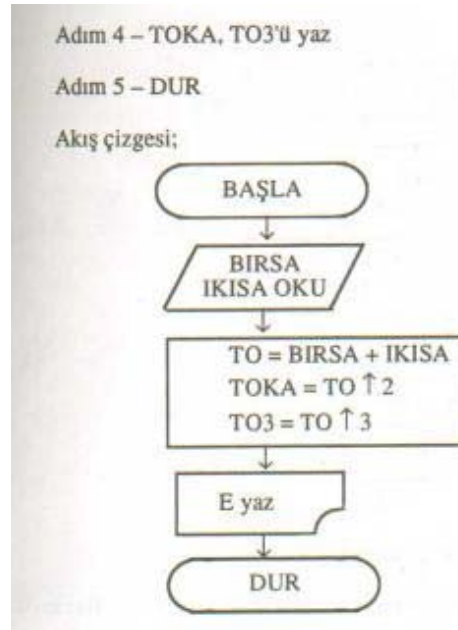
#### Değişkenler

BIRSA: Birinci sayıyı,  
IKISA: İkinci sayıyı,  
TO: Toplamı,  
TOKA: Toplamın karesini,  
TO3: Toplamın küpünü gösterir.

#### Algoritma

Adım 1-Başla  
Adım 2-BIRSA.ve IKISA'yı oku  
Adım 3-TO=BIRSA+IKISA  
TOKA=TO<sup>2</sup>  
TO<sup>3</sup>  
Adım 4-TOKA,TO3'ü yaz  
Adım 5-DUR

#### Akış Şeması



ÖRNEK 5:Terminalden okunan bir sayının 10 fazla ve 10 eksiğini bulup yazan programın algoritma ve akış şeması şöyledir.

#### Değişkenler

SAYI=Okunacak sayı

F10=Sayının 10 fazlası

E10=Sayının 10 eksiğini gösterebilir.

#### Algoritma

Adım 1-Başla

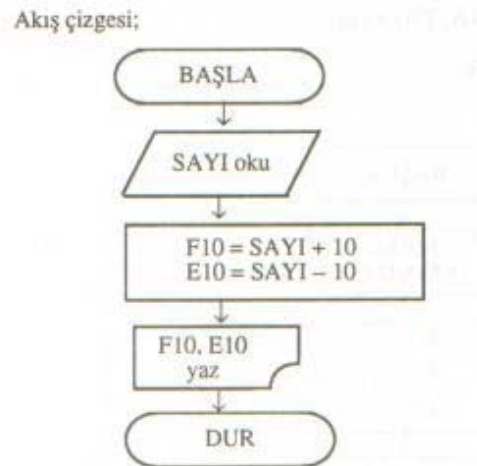
Adım 2-Terminalden SAYI oku

Adım 3-F10=SAYI+10

Adım 4-E10=SAYI-10

Adım 5-DUR

#### Akış Şeması



## MANTIKSAL AKIŞ ŞEMALARI

Geniş ölçüde mantıksal kararları içeren akış şemalarıdır. Hesap düzenleri genellikle basittir.

ÖRNEK 6:A ve B gibi iki sayıdan büyüğünü printerle yazdıran algoritma ve akış şeması şöyledir.

### Algoritma

Adım 1-Başla

Adım 2-A,B'yi oku

Adım 3-A=B ise Adım 7'ye git

Adım 4-A>B ise Adım 6'ya git

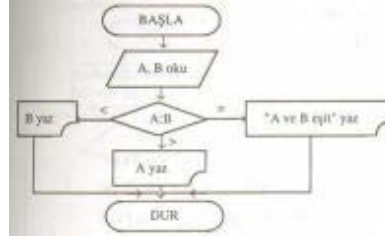
Adım 5-B'yi yaz Adım 8'e git

Adım 6-A'yı yaz Adım 8'e git

Adım 7-"A ve B eşit" mesajını yaz

Adım 8-DUR

### Akış Şeması



ÖRNEK 7:100 kişilik bir sınıfta 18-22 arası her yaş grubunda kaç öğrencinin bulunduğunu belirleyip yazan programın akış çizelgesi şöyledir.

### Değişkenler

OSA:Öğrenci sayısını,

I18:18 yaşında olan öğrencilerin sayısını

I19:19 yaşında olan öğrencilerin sayısını

I20:20 yaşında olan öğrencilerin sayısını

I21:21 yaşında olan öğrencilerin sayısını

I22:22 yaşında olan öğrencilerin sayısını

IYOS:İşlem yapılan öğrenci sayısını gösterebilir.

### Algoritma

Adım 1-Başla

Adım 2-I18=0 I19=0 I20=0 I21=0 I22=0 IYOS=0

Adım 3-OSA oku

Adım 4-OSA=18 ise I18'i arttır, Adım 9'a git

Adım 5-OSA=19 ise I19'u arttır, Adım 9'a git

Adım 6-OSA=20 ise I20'yi arttır,Adım 9'a git

Adım 7-OSA=21 ise I21'i arttır,Adım 9'a git

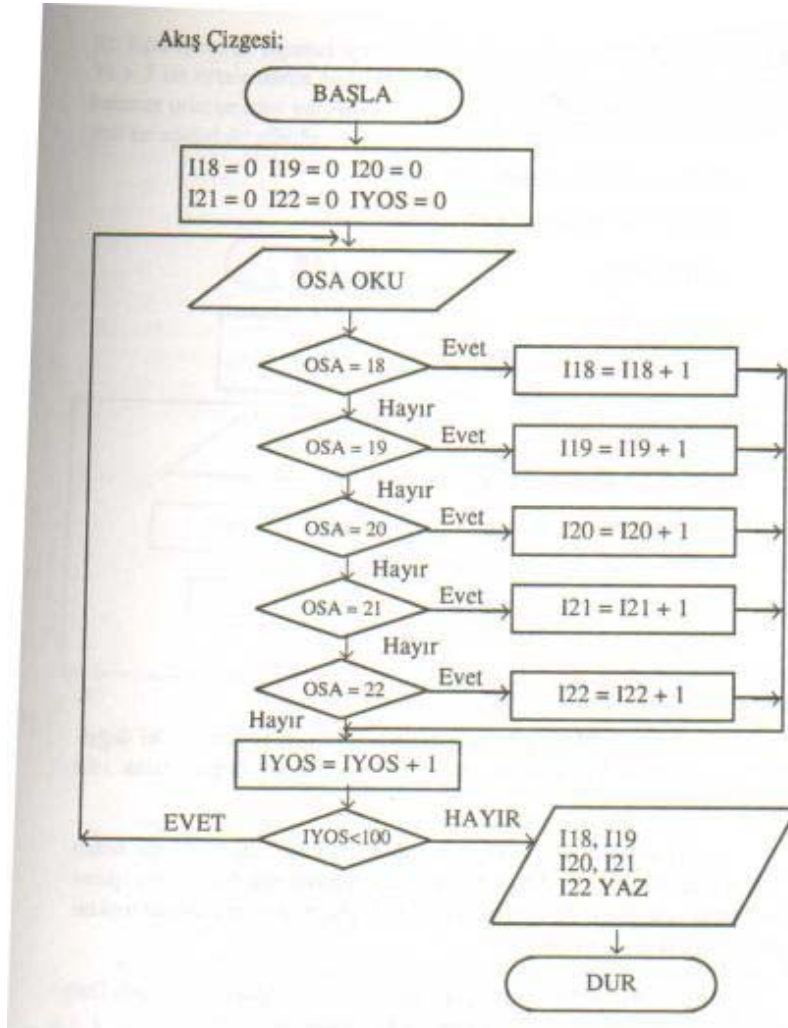
Adım 8-OSA=22 ise I22'Yİ arttır,Adım 9'a git

Adım 9-IYOS<100 ise Adım 3'e git

Adım 10-I18, I19, I20, I21, I22, yaz

Adım 11-DUR

### Akış Şeması



### YİNELİ (İTERATİF,ÇEVİRİMLİ,DÖNGÜLÜ)AKIŞ ŞEMALARI

Sorunun çözümü için,çözümde yer alan herhangi bir adım ya da aşamanın birden fazla kullanıldığı akış şemalarına denir.İş akışları genel olarak giriş ya da başlangıç değeri verme,hesaplama,kontrol biçimindeölmaktadır.

ÖRNEK 8:3 öğrencinin bir sınavdan aldıkları notların ortalamasını bulan v yazan bir programın algoritma ve akış şeması şöyledir.

ÖRNEK 7:100 kişilik bir sınıfta 18-22 arası her yaş grubunda kaç öğrencinin bulunduğunu belirleyip yazan programın akış çizelgesi şöyledir.

### Değişkenler

ONOT:Öğrencinin notunu,

INOT:Notların toplamını,

NORT:Notların ortalamasını,

ISAYI:Öğrenci sayısını gösterebilir.



**Algoritma****Adım 1-Başla****Adım 2-INOT=0****Adım 3-ISAYI=0****Adım 4-ONOT oku****Adım 5-INOT=INOT+ONOT****Adım 6-ISAYI=ISAIY+1****Adım 7-ISAYI<3ise Adım 4'e git****Adım 8-NORT=INOT/3****Adım 9-NORT YAZ****Adım 10-DUR**

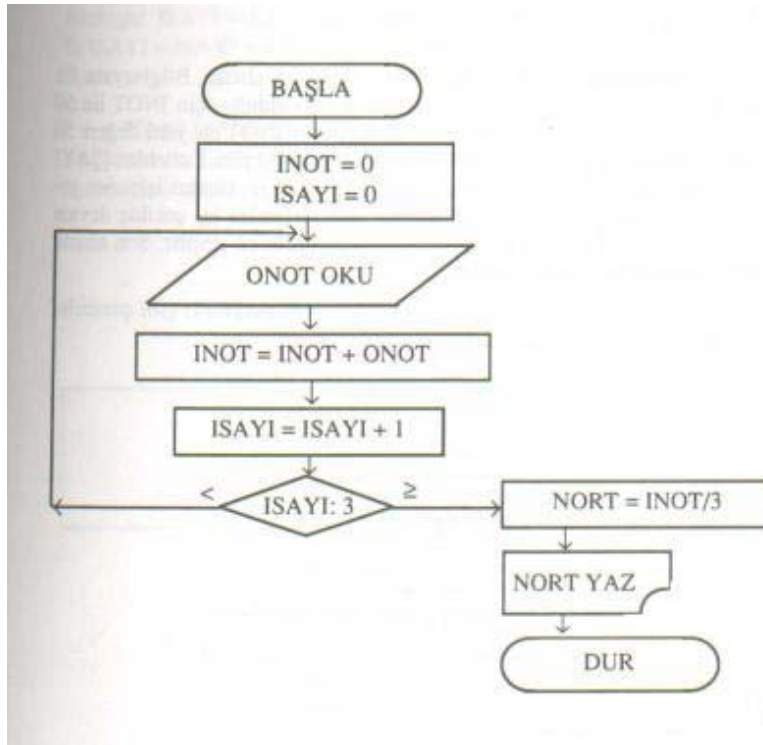
Görüldüğü gibi,Adım 2 ve 3'te INOT ve ISAYI ismi ile iki değişken için bellekte yer ayrılmış,ayrılan yerlerde ilk değer olarak sıfır atanmıştır.

Adım 4'te herhangi bir öğrencinin sınavdan almış olduğu notun değeri okutulmaktadır.

Adım 5'tenotların toplamının bulunması işlemi yer almaktadır.İşlemde ONOT,INOT ile toplanmakta ve bulunantoplama INOT'a aktarılmaktadır.

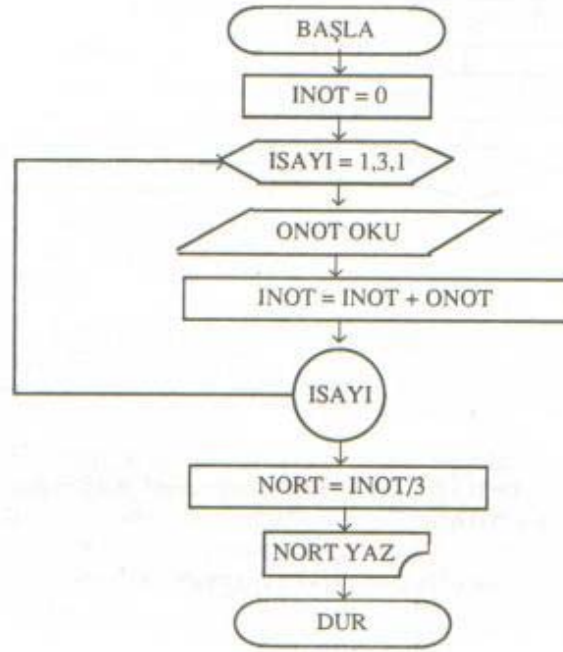
Adım 6'da öğrenci sayısını gösteren ISAYI değişkeninin değeri,"bir öğrenci için işlem yapıldı"anlamında 1 arttırılmaktadır.

Adım 7'de ISAYI'nın değeri 3(toplam öğrenci sayısı) ile karşılaştırılmaktadır.Eğer sayı<3 ise Adım 4'e dönülmektedir.Eğer işlem 3 öğrenci içinde yapılmışsa yani ISAYI=3 ise ya da ISAYI>3 ise ortalamanın hesaplandığı Adım 8'e geçilmektedir.Adım 9 da,bulunan ortalamanın yazılması ile ilgilidir.

**Akış Şeması**

Akış şeması incelendiğinde not okuma, toplam hesaplama ve öğrenci sayısını 1 arttırma işlemlerinin 3 kez tekrarlandığı anlaşılmaktadır.Eğer program 500 öğrenci için yapılmış olsaydı,sözü edilen işlemler 500 kez tekrarlanacaktı. Üç öğrencinin notlarının, sırası ile 50,60 ve 70 olduğunu varsayalım.Bu durumda akış şemasının işlemesi şöyle olacaktır

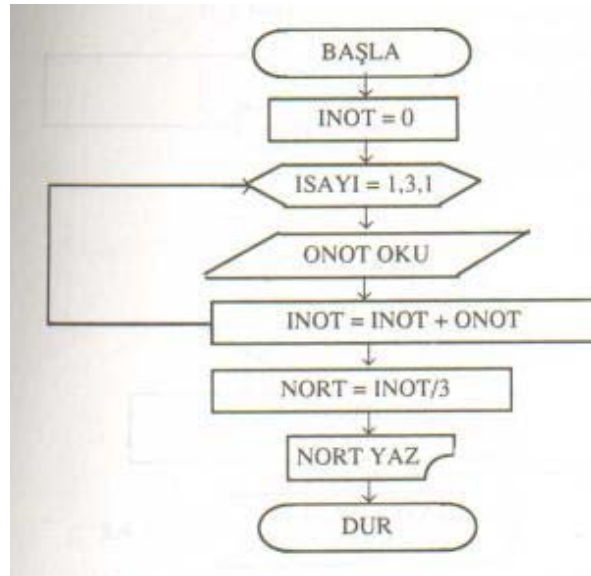
**Akış Şeması**



Başlangıçta INOT ve ISAYI "0" değerini alırlar. Bilgisayara ilk not olarak 50 okutulur. Başlangıçta INOT=0 olduğu için INOT ile 50 toplanır ve sonuç INOT'a aktarılır. Dolayısıyla INOT'un yeni değeri 50 olur. Daha sonra ISAYI'nın değeri INOT'ta olduğu gibi 1 arttırılır. ISAYI=1 olduğu için 3 ile karşılaştırma yapıldığında tekrar okuma işlemine geri dönlür ve ikinci not olarak 60 okutulur. İşlemler bu şekilde devam eder. ISAYI=3 olunca ortalamanın hesaplamasına geçilir. Son olarak bulunan ortalama yazılır ve durulur.

Aynı sorun yineleme sembolü kullanılarak şöyle yazılabilir.

#### Akış Şeması



ÖRNEK 9:

N sayısını ekrandan okutarak faktöriyelini hesaplayan ve yazan akış şeması şöyledir.

#### Değişkenler

**NFAK=N faktöriyel (N!) değerini,**

**ISAYI=1'den N'e kadarsayıları göstereceksin,**

**NFAK=1\*2\*.....\*N**

**Algoritma**

Adım 1-Başla

Adım 2-N'i ekrandan oku

Adım 3-NFAK=1

Adım 4-ISAYI=1

Adım 5-ISAYI=ISAYI+1

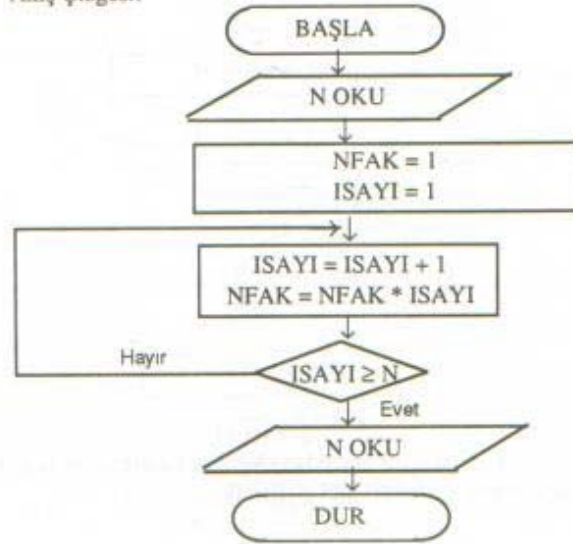
Adım 6-NFAK=NFAK\*ISAYI

Adım 7-Eğer ISAYI Adım 8-NFAK yaz

Adım 9-Dur

**Akış Şeması**

Akış Çizgesi:



Aynı soru yineleme sembolü kullanılarak aşağıdaki gibi de çizilebilir.

ÖRNEK 9: Klavyeden girilen, bir öğrencinin numarasını, ismini ve bilgisayar programlama dersinin 3 vize sınavından aldığı notları okuyan, bu notların aritmetik ortalamasını bulan, eğer ortalaması 50'ye eşit veya 50'den büyükse yazıcıya numara, isim, notlar ve vize notlarının ortalamasını, küçükse numara, isim ve "tekrar" mesajı yazan programın algoritma ve akış şeması şu şekildedir.

**Değişkenler**

INO: öğrencinin numarasını,

AD: öğrencinin ismini,

VIZE1: 1. vize sınavını,

VIZE2: 2. vize sınavını,

VIZE3: 3. vize sınavını,

VIZORT: üç vize sınavının aritmetik ortalamasını,

VIZTO: üç vizenin toplamını gösterebilir.

**Algoritma**

Adım 1-Başla

Adım 2-INO,AD,VIZE1,VIZE2,VIZE3,oku

Adım 3-VIZTO=(VIZE1+VIZE2+VIZE3) ve VIZORT=VIZTO/3 bul.

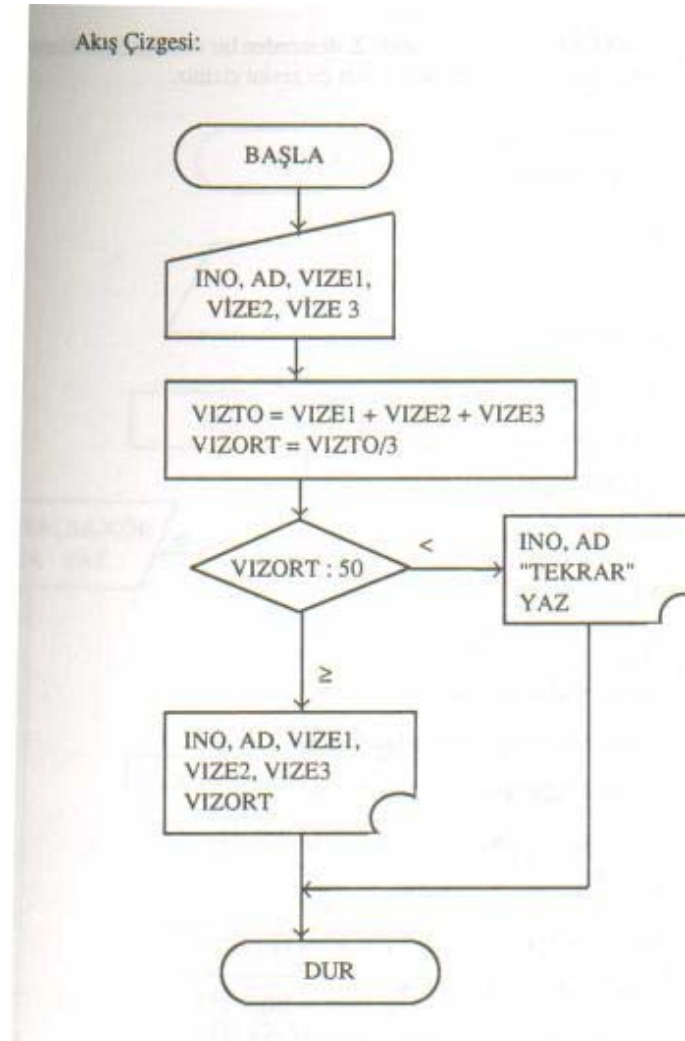
Adım 4-Eğer VIZORT $\geq$ 50 ise Adım 6'ya git.

Adım 5-INO,AD ve "TEKRAR" yaz ve Adım 7'ye git.

Adım 6-INO,AD,VIZE1,VIZE2,VIZE3,VIZORT yaz

Adım 7-DUR

### Akış Şeması



ÖRNEK 10: Bir okulda bulunan 10 sınıftaki 30'ar öğrencinin herbirinin 12 şer dersten aldıkları notların ortalamasını bulan ve öğrenci ismi ile not ortalamasını yazan programın, algoritması ve akış şeması şöyledir:

### Değişkenler

ISS: Sınıf sayacı,

IOS: Öğrenci sayacı,

DN: Ders notu,

DNS: Ders notu sayacı,

DNT: Ders notlarının toplamı,

DNO:Ders notlarının ortalaması,

OGAD:Öğrencinin adı

### Algoritma

Adım 1-Başla

Adım 2-SS=0

Adım 3-IO=0

Adım 4-DNS=DNT=0

Adım 5-OGAD oku

Adım 6-DN oku

Adım 7-DNS=DNS+1(ders notu sayacı 1 artır.)

Adım 8-DNT=DNT+DN(notları topla)

Adım 9-Eğer DNS<12 ise adım 6'ya git

Adım 10-DNO=DNT/12(ortalamayı hesapla)

Adım 11-OGAD,DNO yaz

Adım 12-IO=IO+1(öğrenci sayacını bir artır)

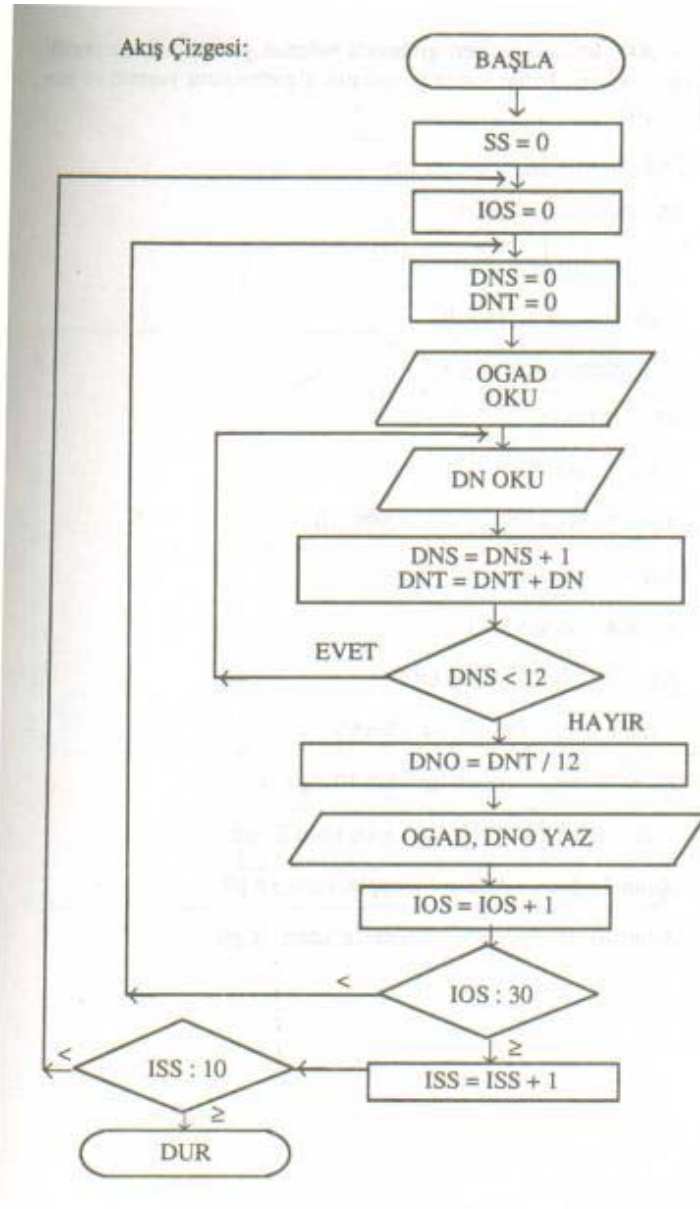
Adım 13-Eğer IO<30 ise adım 4'e git

Adım 14-ISS=ISS+1(sınıf sayacını bir artır)

Adım 15-Eğer ISS<10 ise adım 3'e git.

Adım 16-Dur

### Akış Şeması



ÖRNEK 11:300 elemanlı bir veri grubunda bulunan pozitif,sıfır ve negatif değerlerin sayısını bulup yazan programın algoritması ve akış şeması şöyledir:

#### Değişkenler

SS:Okunan sayı adedi

PSS:pozitif sayı adedi

NSS:Negatif sayı adedi

SSS:Sıfır sayı adedi

#### Algoritma

Adım 1-Başla

Adım 2-SS=PSS=NSS=SSS=0

Adım 3-Sayı oku

Adım 4-SS=SS+1

Adım 5-Eğer SS>300 ise dur

Adım 6-Eğer  $sayı < 0$  ise adım 9'a git

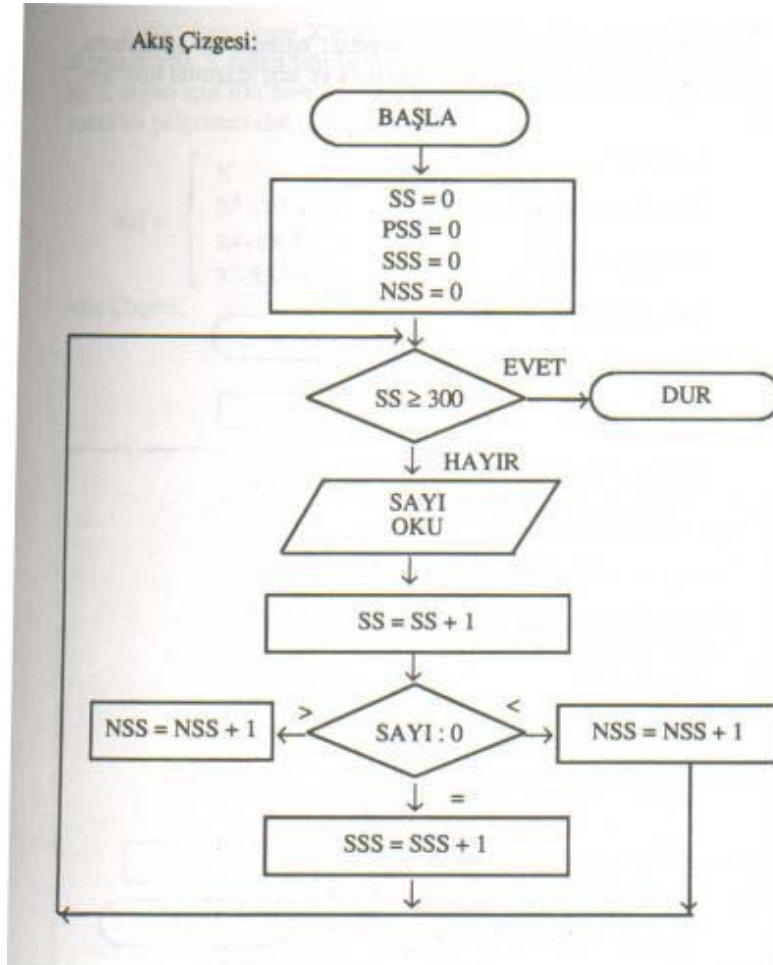
Adım 7-Eğer  $sayı = 0$  ise adım 10'a git

Adım 8- $PSS = PSS + 1$  hesapla, adım 3'e git

Adım 9- $NSS = NSS + 1$  hesapla, adım 3'e git

Adım 10- $SSS = SSS + 1$  hesapla, adım 3'e git

### Akış Şeması



ÖRNEK 12: Bir sınıfta bulunan belirsiz sayıda öğrencilerin numarasını, adını ve soyadını yazan programın algoritması ve akış şeması şöyledir.

### Değişkenler

JNO: Öğrencinin numarası

AD: Öğrencinin adı

SAD: Öğrencinin soyadı

### Algoritma

Adım 1-Başla

Adım 2-Eğer okuma bittiyse Adım 6'ya git

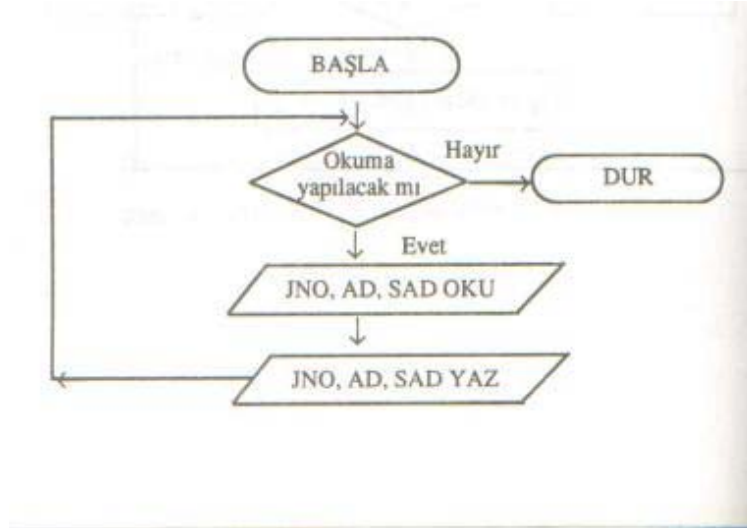
Adım 3-JNO, AD, SAD oku

Adım 4-JNO,AD,SAD yaz

Adım 5-Adım 2'ye git

Adım 6-Dur

#### Akış Şeması



ÖRNEK 13:f(x) fonksiyonu x'in çeşitli değerlerine göre aşağıdaki şekilde değişmektedir.x değeri sıfır ile 5 arasında 0.5 aralıkla arttığına göre her bir x değeri için f(x) fonksiyonunu hesaplayan ve x ile f(x) değerlerini yazan programın akış şeması şöyledir.

#### Akış Şeması

