

## ALGORİTMA

Algoritma; her tür ayrıntı göz önüne alınarak bir problemin çözüm yönteminin adım adım yazılmasıdır. Bu bir düzen içinde yapılır. Maddeler halinde sıralayacak olursak;

- 1- Algoritmanın bir başlangıcı ve bir sonu olmalıdır.
- 2- İşlem akışı aksi belirtilmediği taktirde yukarıdan aşağıya doğrudur.
- 3- Her bir adımda tek komut ya da tek işlem yazılmalıdır.
- 4- Girdiler veya çıktılar bulunmalıdır.
- 5- Her işlem adımının bir satır numarası olmalıdır
- 6-
- 7- Bir algoritma içerisinde tekrarlanan işlemler varsa bu işlemleri ayrı ayrı yazmamıza gerek yoktur. Tekrarlanan işlemi bir kez yazıp bu işlemi yeniden yapacağımızda işlemin olduğu satır numarası vererek işlemin yapılacağı adıma geri dönüş sağlanabilir.
- 8- Aritmetik işlemlerde kullanılan = in anlamı sağ taraftaki işlemlerin yapılıp sol taraftaki değişkene atanması anlamındadır.

## Değişken

Bilginin bilgisayar belleğinde saklanması olayıdır. Temsili olarak bilginin saklandığı alana hücre denilir. Bu hücreye bir isim verilir ve bu isimle anılır.

Kağıdınıza matris şeklinde küçük kutucuklar çizelim. Bu kutucukların içerisine sayı, harflerden ya da her ikisinden oluşan bilgiler yazalım.

Ayşe	Kerem	23
Gül	35	Gül
No:21	23	Atmaca15

Şimdi burada her kutucuğun bir isminin olması gereklidir. Kutucuklarda dikkat edilirse iki adet 23 sayısı, iki adet de Gül ifadesi var. Bu ifadeler farklı anlamlar taşımaktadır. “Gül” ifadelerinden birisi insan ismi iken diğer bir çiçek ismidir, ama hangisi ?

Bu karışıklığı önlemek için kutucukların birer isimleri olsaydı.

			yaş
Ayşe	Kerem	23	
Gül	35	Gül	çiçek
No:21	23	Atmaca15	
adı			
	gün		

Artık kutucukları karıştırmak gibi bir durum söz konusu değildir.

Adı kutucuğunda ne var?

Çiçek kutucuğunda ?

Bugün ayın kaçındır denildiğinde gün kutusu buna cevap verecektir.

## Değişken Türleri

Yukarıda bahsettiğimiz kutucukların içerisine koyacağımız bilginin türüne denilir. Şimdilik iki türlü bilgidten bahsedeceğiz. Bunlar;

**Sayısal bilgi:** 0 ile 9 arasındaki rakamlardan ve sadece nokta işaretinden oluşmaktadır.

**String bilgi (alfasayısal):** Harfler, simgeler, noktalama işaretleri, boşluk ve rakamlardan oluşmaktadır.

**Değişken Tanımlama:**

Algoritma yazmaya başlarken bilgilerin mutlaka bir değişkende saklanması gereklidir. Bundan dolayı hangi tür bilgi saklanacaksa öncelikle bellekte ona yer ayırmamız gerekmektedir.

Örneğin 0.20 litre su taşımak istenirse, bir su bardağını züccaciye den almanız gerekmez mi? Bu suyu taşımak için süzgeç alır mısınız ? Fakat bir miktar nohut için bardak ya da süzgeç alınabilir. O zaman işe başlamadan önce taşıyacağımız nesne için doğru kabı seçmemiz gereklidir. Bu işleme de tanımlama denilmektedir.

Bellekte saklayacağımız bilgi sayısal bilgi ise şu şekilde; örneğin boy bilgisi için  
Boy=0

Kişi adresini bellekte saklayacak olursak ta

Adres="" ya da

Adres="degol cad. no:5"

Şekilde yazılarak iki değişken için bilgisayar belleğinde yer ayırt etmiş oluruz. Burada yapılan işlem bir kutucuk olsun adı boy, bilgi türü de sayısal olsun denilmiştir. Sayısal bilgiler tırnaksız, string bilgiler tırnaklı olarak yazılır.

**Değişkene değer atama**

Algoritmada bir değişken tanımlanırken aynı zamanda da içerisine de ilk değeri atanmış olmaktadır. Boy=0 denildiğinde sayısal bir değişken ve aynı zamanda da içerisinde sıfır olsun denilmiştir.

Boy=0

Boy=25

Ardı ardına iki satır yazıldı, bunlardan ilki değişkeni tanımlama ve değer atama, ikincisi de onun değerini değiştirme yani atama yapılmış oldu.

**Değişken isimlendirme:**

Değişkenlere isim verilirken bazı kurallara uyulması gereklidir. Bu kurallardan bazıları programcılığın bazıları da daha düzenli program yazabilmek için bizim koyacağımız kurallar olacaktır.

- 1) Türkçe karakter kullanılmaz . orn:İ,Ş,Ö,Ç...
- 2) Özel karakter kullanılmaz. :, \$, !, ^, (, ) ...
- 3) sadece \_ kullanılabilir. Örnek: sayi\_1
- 4) değişken ismi içindeki bilgiyi anımsatmalı.

**Örnek:** s="Demircan" → yanlış isimlendirme  
soyadi="Demircan" → doğru

5) Eğer birden fazla kelimeden oluşuyorsa ilk kelime hariç diğer kelimelerin ilk harfleri büyük olmalı.

**Örnek:**

ogrdogumyeri-->yanlış isimlendirme

ogrDogumYeri-->dogru isimlendirme

**İlişkisel operatörler (kıyaslama)**

küçüktür	<
büyüktür	>
büyük yada eşit	>=
küçük yada eşit	<=
eşit	=
eşit değil	<>

**Koşul İfadeleri**

Algoritma yapılarında çoğu zaman belli koşullar sağlandığı sürece geçerli olan durumlar vardır. Bu günlük hayattaki problemlerde de böyledir. Örneğin yağmur yağıyorsa şemsiyeni aç. Bu tip ifadeler

Eğer koşul ise işlem

Eğer koşul ise işlem1 değilse işlem2

gibi kullanılır. Bu, programlama dillerinde IF deyimine karşılık gelir.

“Eğer” den sonraki koşulun sonucu doğru ise “ise” den sonraki, yanlış ise “değilse” den sonraki işlem yapılır.

**Örnek:**

Sayı=5                      sayı kutucuğuna 5 değeri konulmuş olsun

Eğer sayı>3 ise x=5

Burada sayı değişkeni içerisinde 5 olduğu için sayı>3 durumu da sağlandığı için x değişkenin yeni değeri artık 5 olmuştur.

**Örnek:**

Sayı=5

Eğer sayı>6 ise x=25 değilse x=20

Burada sayı değişkeni, 6 sayısından büyük olmadığı için x in yeni değeri 20 olacaktır.

**Mantıksal Bağlaçlar**

Birden fazla koşul ifadesi var ise, bunlar ve/veya bağlaçları kullanılarak bağlanırlar.

Bir kişiye birden fazla soru yöneltecekseniz, bu durumda sorularınız arasına “ve” “veya” bağlaçları konulması gerekir.

**Örnek:**

Tek soru sorulursa: Hava sıcaklığı 25 den fazla mı ? Beklenen cevap ya EVET, yada EVET DEĞİL olmalıdır. Algoritma diliyle sorumuz aşağıdaki olmalıdır.

Eğer sıcaklık>25 ise

İki soru sorulursa: Hava sıcaklığı 25 den fazla ve nem 60 ın üstünde mi ? Burada dikkat edilirse iki soru “ve” bağlacıyla birleştirildi. Beklenen cevap her bir soru için ayrı olacak, sonuçta her iki cevabın da birleştirilerek tek cevabı olmalıdır. Algoritma diliyle

Eğer sıcaklık>25 ve nem>60 ise

Şeklinde yazılır.

Her zaman “eğer” ifadesinde koşulların sonucunda bir tane cevabı olmalıdır. Bu ya EVET dir, yada EVET DEĞİL dir.

**Örnek:**

Bir kişinin kilosu 60 dan büyük ve 80 den küçük ifadesinin karşılığı

Eğer 60<kilo<80 ise ...                      şeklindeki yazım yanlıştır.

Eğer kilo>60 ve kilo<80 ise ...                      şeklindeki yazım doğrudur.

**Örnek:**

Bir öğrencinin vize notu 50 veya 55 ise ifadesinin karşılığı  
 Eger vize=50 veya 55 ise ... şeklindeki yazım yanlıştır.  
 Eger vize=50 veya vize=55 ise ... şeklindeki yazım doğrudur.

**Algoritmada matematiksel işlem**

Toplama +, çıkarma -, bölme / , çarpma \*, üs alma ^ işaretleriyle gösterilir.

**Örnek:**

Sonuc=2\*3+1/2  
 x=2^4

işleminde 8.5 değeri sonuc değişkenine atanır.  
 işleminde x değişkenine 16 değeri atanır.

**Sayac tekniği**

Bir değişken, kendisinin önceki değeri üzerine bir sayı eklenerek oluşan yeni değerinin yine kendisinde saklanması istenir. Bu tekniğe **sayac** tekniği denilir.

Değişkenimize sayac ismini verelim.

Sayac=10 denildiğinde sayac değişkeninin değeri 10 olmuş olur.

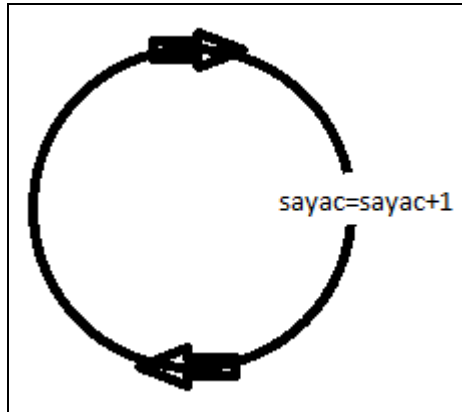
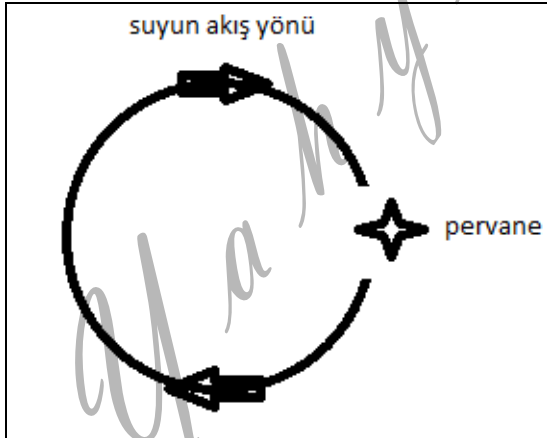
Sayac=11 denildiğinde yeni değeri 11 olur. Peki biz sayac içindeki değere 1 ekleyerek yeni değerini elde etmek istersek;

Sayac+1 denildiğinde, sayacın önceki değeri 11 idi, bu değere 1 ekledik ve 12 sayısını elde ettik. Bu 12 sayısını da bir değişkende saklamak istiyoruz, bunun adı toplam olsa

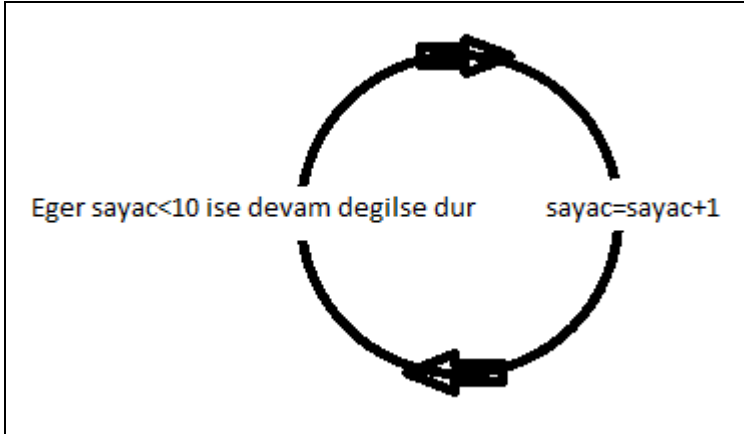
Toplam=sayac+1 yazmalıyız. Toplam değişkeninin değeri 13 oldu.

Peki bu 13 sayısını toplam değişkeninde değil de sayac değişkeninde saklamak istersek, eşit işaretinin sol tarafına sayac yazmamız yeterli olacaktır.

$sayac=sayac+1$  artık sayac değişkeninin yeni değeri 13 olmuş olur. Burada matematiksel olarak düşünülmemelidir. Zira böyle bir eşitlik söz konusu değildir. Bu yazım tekniğine sayac tekniği denilir.



Soldaki şekilde görüldüğü gibi suyun pervanenin kanatlarına çarparak oluşan dönme eylemi, su sayacınızın artmasına sebep olacaktır. Sağdaki şekilde ise programın her işlemi yaptığında sayacınızın değeri hep bir artacaktır. Bu iki döngüsel harekette, sayac=0 başlangıç değeri olmak kaydıyla, araya bir koşul koyabilir miyiz?



Bu döngüsel eylemde bir tur atılacak, sonra sayac'ın değeri kontrol edilecek, koşul sağlanıyorsa dönmeye devam edilecek, ne zaman ki sayac'ın değeri 10 olduğu anda işlem duracaktır.

İlk tur : sayac=sayac+1 işleminin sonucu 1, peki  $1 < 10$  mü, evet dönmeye devam,

İkinci tur : sayac=sayac+1 işleminin sonucu 2, peki  $2 < 10$  mü, evet dönmeye devam,

...

Dokuzuncu tur: sayac=sayac+1 işl. Sonucu 9, peki  $9 < 10$  mü, evet dönmeye devam,

Onuncu tur : sayac=sayac+1 işl. Sonucu 10, peki  $10 < 10$  mü, hayır dönme duracak.

Dikkat edilirse 10 kez tur atılmış oldu.

Sayaç tekniğinde, matematiksel operatöre göre isimlendirilir. + olursa toplama sayacı, \* (çarpı) olursa çarpım sayacı olarak isimlendirilir.

Örnek:

Carpim=carpim \* 3

Bu işlemde 3 ve katları elde edilir.

Sayi=sayi - 10

Bu işlemde 10 ar eksiltilir.

### Algoritmada çıktı işlemi

Algoritmada sabit bir string ifade yada hesap edilen sonuçların ekrana yazdırılmasına **çıkı** işlemi denilir. Bu işlem için kullanılan komut **YAZ** dir.

Örnek:

YAZ "Selam"

ifadesinin sonucunda ekranda **Selam** ifadesi görüntülenir.

Örnek:

YAZ 3+4

adım işleme tabi tutulur.

ifadesinin sonucunda ekranda **7** ifadesi görüntülenir ve sıradaki

### Algoritmada girdi işlemi

Algoritmada kullanıcının klavyeden bilgi girmesi isteniyorsa buna **girdi işlemi** denilir. Bu işlem için kullanılan komut **GIR** dir.

Örnek:

Gir sayı ifadesi işleme alındığında klavyeden bir bilgi girilmesi ve ardından da ENTER tuşuna basılması için kullanıcı beklenir. ENTER dan önce girilen bilgi ilgili değişkenin içine aktarılır ve sıradaki adım işleme tabi tutulur.

**Soru:** Klavyeden girilen iki sayının toplamını ekrana yazdırınız.

Çözüm:

A10: Basla

A20: sayi1=0

A30: sayi2=0

A40: gir sayi1

A50: gir sayi2

A60: yaz sayi1+sayi2

A70: son

Bu soruda kullanıcıdan iki sayı istenmektedir. Bu nedenle 20 ve 30. adımda iki sayısal değişken tanımlanmıştır. A40'a gelindiğinde beklenir, kullanıcı 5 sayısını tuşlar ve ardından enter'a basılır, A50'ye gelinir yine kullanıcı klavyeden 9 tuşuna ardından enter'a basar. Otomatik olarak sıradaki adım A60'dır. Bu adımda yapılması gereken iki değişkenin içinde bulunan değerler toplanır ve ekrana yazdırılır. 5 ve 9 toplanır ve ekranda 14 sayısı görüntülenir.

**Soru:** Klavyeden girilen öğrenci ismini ekrana yazdırınız?

Çözüm:

A10: Basla

A20: isim=""

A30: Gir isim

A40: Yaz isim

A50: son

Bu soruda istenilen sadece klavyeden bir isim girilecek ve onu ekrana yazdıracak algoritmadır. 20. adımda değişken tanımlanmıştır, 30'da klavyeden Yahya yazılmış ve ardından enter'a basılmıştır. Otomatik olarak program bir alt satır olan A40'a gelmiştir. Bu adımdaki komut ise isim değişkeninin içinde ne varsa onu ekrana görüntüle demektir ve ekranda Yahya ifadesi görüntülenecektir.

**Soru:** Klavyeden girilen 5 sayıyı toplayarak ekrana yazdırınız.

Çözüm:

A10: Basla

A20: sayi1=0

A30: sayi2=0

A40: sayi3=0

A50: sayi4=0

A60: sayi5=0

A70: Gir sayi1

A80: Gir sayi2

A90: Gir sayi3

A100: Gir sayi4

A110: Gir sayi5

A120: toplam= sayi1 + sayi2 + sayi3 + sayi4 + sayi5

A130: Yaz toplam

A140: Son

Dikkat edilirse çözüm biraz uzun oldu. Soruda 5 sayı denilmektedir, ya bu 100 sayı olsaydı ! Bu çözümde benzer satırlar yer almaktadır. 5 tane sayı değişkeni tanımlı, 5 tane de gir ifadesi var.

Bu tür sorular için klasik örneğimiz olan sürahi ve bardak olayı ile anlamaya çalışalım.

**Senaryo 1:** 5 adet bardak ve sürahi alınır. 5 adet bardak doldurulur ve sürahiye boşaltılır. veya

**Senaryo 2:** Bir bardak ve sürahi alınır, 5 kez bardağı doldur boşalt yaparak sürahide 5 bardak suyun birikmesi sağlanır.

Bu iki senaryoyu sınıfta yorumlayalım ...

Böylesi durumlarda bir değişkenle işlemin tekrar edilmesi programcılık mantığı ile uyushmaktadır.

Tekrarlama işleminde **eger** ve **git** deyimleri kullanılacaktır. “Eger” ile mukayese edeceğiz, isteğimiz gerçekleşmedi ise git komutu ile tekrarlanması gereken işlemlerin baş tarafına göndereceğiz. Çözümü yeniden yazalım;

A10: Basla

A20: sayi=0

A30: sayac=0

A40: toplam=0

**A50: sayac=sayac+1**

**A60: Gir sayi**

**A70: toplam=toplam + sayi**

**A80: Eger sayac<5 ise Git A50**

A90: Yaz toplam

A100: Son

Bu çözümde 5 sayı değil de 100 sayı dahi olsa tek yapılması gereken A80’de 5 yerine 100 yazmak yeterli olacaktır.

Burada tekrar edilmesi gereken satır A60 ve A70 dir. Tekrarlama işleminin kaç kez yapılması gerektiğini A50 saymaktadır. Tekrarlama işlemini kontrol edecek satır A80 dir. Burasını sınıfta biraz daha açalım.

**Soru:** 1 den 1000 a kadar olan sayıların toplamını bularak ekrana yazdıran algoritma yazınız.

**Çözüm:** Bu soruda 1 ile 1000 arasındaki sayılar kullanılacaktır. Bu sayılar ardışık ve elde edilmesi kolay olduğundan klavyeden girilmeyecektir. Aksine klavyeden girdirmek hatalı olacaktır. Düşünün; birisi sizden 1000 e kadar tek tek sayar mısınız şeklinde bir istekte bulunsun. Siz 1, 2, 3, ... şeklinde saymaya başlar mısınız, yoksa karşınızdaki kişiye her defasında bir sayı söyleyeyim mi dersiniz !!!!

A10: Basla

A20: sayac=0

A30: toplam=0

**A40: sayac=sayac + 1**

**A50: toplam = toplam + sayac**

**A60: Eger sayac<1000 ise Git A40**

A70: Yaz toplam

A80: son

Burada tekrarlama işleminin başlangıcı A40, kontrolü de A60 da yapılmaktadır. Bu işlemi sınıfta tartışalım.

**Soru:** Bir ile 1000 arasındaki sayılardan çift olanları ekrana yazdırınız.

**Çözüm:** Bu soruda da klavyeden sayı girilmeyecektir. Çünkü sayac mantığıyla bu sayıları elde edilebilmektedir. Sayac değişkeni sıfırdan başlayarak 2'şer artımlarla ilerlerse, çift sayılar elde edilmiş olur.

A10: Basla

A20: sayac=0

A30: sayac=sayac + 2

A40: yaz sayac

A50: Eger sayac<1000 ise Git A30

A60: Yaz toplam

A70: son

**Soru:** Klavyeden girilen 100 adet sayı içerisinde tek olanlar, çift olanlar ve sıfır olanların adetlerini bularak ekrana yazdırınız.

**Çözüm:** Bu soruda klavyeden tek tek sayılar girilecektir, her girişten sonra bu sayı kontrol edilecek, hangi gruba giriyorsa onun adedi bir artırılacaktır. Bu işlem 100 tane sayı girilene kadar devam edecektir. ^ işareti (üs alma) yardımıyla -1 'in üssüne yazılan sayı tek olursa, sonuc negatif, aksi durumda pozitif olacaktır.

A10: Basla

A15: sayi=0

A20: sayac=0

A21: tsayac=0                      tek sayıları sayması için

A22: csayac=0                      çift sayıları sayması için

A23: ssayac=0                      sıfırları sayması için

A30: Gir sayi

A35: sayac=sayac + 1

A40: Eger (-1)^sayi<0 ise tsayac=tsayac+1

A41: Eger (-1)^sayi>0 ise csayac=csayac+1

A42: Eger sayi=0 ise ssayac=ssayac+1

A50: Eger sayac<1000 ise Git A30

A60: Yaz csayac, tsayac, ssayac

A70: son

### Fonksiyonlar

Programlama dillerinde, sık kullanılan işlemler bir modül haline getirilmiştir. Bu modüle; procedure, alt program, subroutine, fonksiyon, action, event gibi farklı isimler verilmektedir. Bizde bu tanımlara bir ilave yapalım. Bilim yada iş dünyasında bazı işler rutin hale gelmiştir. Bu rutin işlemleri doğru ve hızlı şekilde, her kullanan kişiye aynı sonucu vermesi önemli özellikleri arasındadır. Basit bir örnek vermek gerekirse; pasta ustası her gün kek yapmaktadır. Yıldız şeklindeki keki el yordamıyla yapması mı, yoksa metal bir yıldız kalıbı kullanması mı daha doğru olacaktır. Demek ki, sık kullanılacak işlemler kalıp haline getirilir, “bir kez yaz, defalarca kullan”.

Kullanacağımız programlama diline özgü yüzlerce hazır fonksiyonlar bulunmaktadır. Bu derste bunlardan bazıları kullanılacaktır.

Mid(), len(), val(), str(), \, & fonksiyonları kullanılacaktır.

**Mid():** Metinden parça almaya yarayan bir fonksiyondur. Bu fonksiyon 3 değer alır, geriye string değer verir.

Mid(metin,m,n)                      metin: string ifade, m başlangıç sayısı, n: adet



**Örnek:**

Mid("Oya top oyna",2,2)  
Mid("Yahya",2,3)

sonuç "ya" olacaktır.  
sonuç "ahy" olacaktır.

**Örnek:**

isim="Yahya Demircan"  
mid(isim,1,7)

sonuç "Yahya D" olacaktır. Boşlukta bir karakterdir.

**Örnek:**

i=3  
k=11  
mid("Ankara Üniversitesi",i,k)

sonuç "Ankara Üniv" olacaktır.

**Len()** : String ifadenin kaç karakterden oluştuğu bulan fonksiyondur. String ifadede boşluk karakterinin de sayıldığı unutulmamalıdır.

**Örnek:**

Len("Ali fiş topla")

sonuç 13 olacaktır.

**Örnek:**

Adres="degol cad. No:5 tandoğan"  
Len(adres)

sonuç 24 olacaktır.

**Soru:** Klavyeden girilen öğrenci isminin ilk 3 karakterini ekrana yazdırınız.

Çözüm:

A10: Basla  
A20: isim=""  
A30: gir isim  
A40: Yaz mid(isim,1,3)  
A50: son

**Soru:** "Ankara" kelimesinin harflerini alt alta şekildeki gibi yazdırınız.

Çözüm:

A10: Basla  
A20: sehir="Ankara"  
A30: sayac=sayac+1  
A40: Yaz mid(sehir,sayac,1)  
A50: Eger sayac<len(sehir) ise git A30  
A60: son

A
n
k
a
r
a

Yaz komutu her çalışmada alt alta yazdırır.

**Soru:** Klavyeden girilen cümledeki kelime sayısını bularak ekrana yazdırınız.

**Çözüm:** Bir cümlede normal şartlarda kelimeler arasında bir boşluk yer alır. Boşluk sayısı bulunursa çözüme ulaşılır.

A10: Basla

A20: cumle=""

A25: kelimeSay=0

A26: uzunluk=0

A30: sayac=0

A40: gir sayac

A50: uzunluk=len(cumle)

A60: sayac=sayac+1

A70: Eger mid(cumle,sayac,1)=" " ise kelimeSay=kelimeSay+1

A80: Eger sayac<uzunlukise git A60

A90: yaz kelimeSay+1

----- neden 1 eklendi ?

A100: son

Val(): Bu fonksiyon rakamlardan oluşmuş string ifadeyi sayısal hale dönüştürür. ,string ifade rakam değilse bunun dışındaki karakterlerden birisi olursa sonuç sıfırdır.

**Örnek:**

Val("321")

sonuç 321 sayıdır.

**Örnek:**

Sayi="344"

Val(sayi)

sonuç 344 sayıdır.

**Örnek:**

Val("onbir")

sonuç 0 sayıdır.

**Örnek:**

Val("Ankara")

sonuç 0 sayıdır.

Str(): Bu fonksiyon sayıyı stringe dönüştürerek string fonksiyonların kullanılmasına olanak sağlar.

**Örnek:**

Str(25)

Sonuç string "25" olmaktadır. Bu sayı değildir.

**Örnek:**

Numara=34534

Str(numara)

Sonuç string "34534" halini almaktadır. Numara

değişkeni halen sayısaldır. Sadece içindeki bilgi alınmış stringe dönüştürülmüştür.

**Soru:** Klavyeden girilen sayının rakamları toplamını ekrana yazdırınız.

**Çözüm:** Sayı stringe dönüştürülür, rakamları tek tek ele alınır, sayıya dönüştürülür ve toplamı bulunur.

A10: Basla

A20: sayi=0

A25: sSayi=""

A26: uzunluk=0

A28: toplam=0

A30: sayac=0

A40: gir sayi

A45: sSayi=str(sayi)

sayi içindeki değer alınır, stringe dönüştürülür

A50: uzunluk=len(sSayi)

A60: sayac=sayac+1

A70: toplam=toplam+val(mid(sSayi,sayac,1))      alınan basamak sayıya dönüştürülür

A80: Eger sayac<uzunluk ise git A60

A90: yaz toplam

A100: son

**\:** Bu işaret bir fonksiyon gibi davranır. Bir sayının diğer bir sayıya bölümünden oluşan bölümün tam kısmını verir. Tam bölme olarak bilinir. Normal / işleminde bölüm kesirlidir.

**Örnek:**

3\2      sonuç 1 dir.

7\3      sonuç 2 dir.

7/3      sonuç 2.33 dür.

**Soru:** İki basamaklı sayılar içerisinde 7'nin katlarını ekrana yazdırınız.

**Çözüm:** Bu sorunun iki farklı çözümü mevcuttur. İki basamaklı sayılar 10 dan başlar 99 kadar devam eder. Burada \ işaretini kullanarak çözüme gidelim.

A10: Basla

A20: sayac=0

A30: sayac=sayac+1

A40: Eger sayac/7=sayac/7 ise Yaz sayac

A50: Eger sayac<99 ise git A30

A60: son

**&:** Bu fonksiyon string ifadeleri birleştirmek için kullanılır.

**Örnek:**

Adres1="Döğol cad"

Adres2="No:5 tandoğan"

Adres= adres1 & adres2

iki bilgi birleştirilip adres değişkenine atanmıştır.

**Soru:** Klavyeden ayrı ayrı girilen ad ve soyadın arasına bir boşluk koyarak ekrana yazdırınız.

**Çözüm:**

A10: Basla

A20: ad=""

A30: soyad=""

A40: butun=""

A50: Gir ad

A60: Gir soyad

A70: butun=ad & " " & soyad

A80: Yaz butun

A90: son

**Soru:** Klavyeden girilen cümlelerin kelimelerini alt alta yazdırınız.

**Çözüm:** Bu çözümde cümledeki boşluklardan yararlanılacaktır.

A10: Basla

A20: cumle=""

A30: uzunluk=0

A40: sayac=0

A50: Gir cumle

A60: sayac=sayac+1

A70: kelime=kelime & mid(cumle,sayac,1)

A80: Eger mid(cumle,sayac,1)="" ise yaz kelime

A90: Eger mid(cumle,sayac,1)="" ise kelime=""

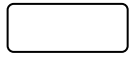
A100: Eger sayac<len(cumle) ise git A60

A110: son

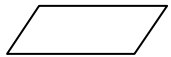
Bu satır neden ?

### Akış Şeması

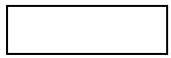
Bir problemin çözüm aşamalarının şekillerle ifade edilmesidir. Akış şemasında kullanılan şekiller aşağıda verilmiştir.



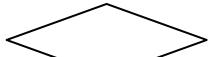
Basla ve son ifadelerinde kullanılır.



Klavyeden bil girişi için kullanılır, Gir komutu yerine geçer.



İşlem ifadelerinde kullanılır.



Karar verme ifadelerinde kullanılır. "Eger" deyimi yerine geçer.



Ekrana çıktı işlemlerinde kullanılır. "Yaz" komutu yerine kullanılır.



Bağlantı noktalarında kullanılır.



Yönlendirmelerde kullanılır.

**Soru:** Bir ile 10 arasındaki tamsayıların karelerini ve kendilerini ekrana yazdıran algoritmanın akış şemasını çizin.

**Çözüm:**

