

# PROGRAMLAMA TEMELLERİ

## BÖLÜM 2

ÖĞR. GÖR. HAKAN CAN ALTUNAY

## ALGORİTMA MANTIĞI

Bir problemin çözümünde izlenen adımlar dizisine algoritma adı verilir. Aslında algoritma, problemlerin çözümünü gerçekleştirebilmesi için bilgisayar öğretilen işlem basamaklarıdır. Bilgisayar, bir problemi çözerken hangi giriş değerlerini kullanacak, bunları işlerken ne tür yöntemler kullanacak, hangi sonuçları üretecek ve bu sonuçları nerede gösterecek veya saklayacak vb. adımların hepsi hazırlanan algoritmanın herhangi bir programlama dilinin kurallarına uygun olarak yazılmış komutlarıyla bilgisayara iletilir. Algoritmanın özel geometrik şekillerle gösterilmesine ise akış diyagramı denir.

Bir problem çözülürken yapılacak ilk iş problemin algoritmasının hazırlanmasıdır. Daha sonra istenilen programlama dili komutlarıyla kodlanır. Bir problemin algoritması hazırlanabiliyorsa, bu problem bütün programlama dillerinde çözülebilir ya da kodlanabilir. Değişecek olan tek ifade kullanılan programlama diline ait kodlardır. Bir program yazılırken aşağıdaki adımlar takip edilir.

- 1- Programın algoritması hazırlanır veya akış diyagramı çizilir.
- 2- Algoritma veya akış diyagramı programlama diliyle kodlanır.
- 3- Program çalıştırılır.
- 4- Yazım hatası varsa düzeltilir.
- 5- Program çalıştığında sonuçlar istenildiği gibi değilse ilk adıma geri dönülür ve algoritma tekrar incelenir.

### PROGRAMLAMA HATALARI

Bir program yazılırken yazım hatası ve mantıksal hata olmak üzere iki farklı hata yapılır. Eğer komut kullanılan programlama dilinin kurallarına uygun yazılmamışsa ortaya çıkan hataya yazım hatası denir. Yazım hatasının düzeltilmesi kolaydır. Fakat programın genel yapısında hata varsa, (örneğin yanlış operatör kullanımı, yanlış denklem kodlanması vb. ) ortaya mantık hatası çıkar. Ve program baştan sona denetlenerek hatalar bulunmaya çalışılır.

### ALGORİTMALARDA KULLANILAN OPERATÖRLER

Genel olarak algoritmalarda kullanılan operatörler, matematiksel, karşılaştırmalı, mantıksal, küme işlem, alfasayısal ve genel işlem operatörleri olmak üzere 6 gruba ayrılır. Aşağıdaki tablolarda bu operatörlerin anlamları ve kullanımı gösterilmiştir.

## 1- MATEMATİKSEL İŞLEM OPERATÖRLERİ

ÜS ALMA	$\wedge$
ÇARPMA	$*$
BÖLME	$/$
TOPLAMA	$+$
ÇIKARMA	$-$
TAM VE ONDALIK KISMI AYIRMA	$.$

## 2- KARŞILAŞTIRMA OPERATÖRLERİ

EŞİT	$=$
EŞİT DEĞİL	$<>$
KÜÇÜK	$<$
BÜYÜK	$>$
BÜYÜK EŞİT VEYA EŞİT BÜYÜK	$\geq, \Rightarrow$
KÜÇÜK EŞİT VEYA EŞİT KÜÇÜK	$\leq, \Leftarrow$

## 3- MANTIKSAL İŞLEM OPERATÖRLERİ

DEĞİL	$'$
VE	$.$
VEYA	$+$

## 4- KÜME İŞLEM OPERATÖRLERİ

BİRLEŞİM	$+$
FARK	$-$
KESİŞİM	$*$
EŞİT	$=$
EŞİT DEĞİL	$<>$
SOLDAKİ KÜME SAĞDAKİNİN ALT KÜMESİ	$=<, \leq$
SAĞDAKİ KÜME SOLDAKİNİN ALT KÜMESİ	$\Rightarrow, \geq$

## 5- ALFASİYASAL OPERATÖRLER

BİRLEŞTİRME	$+$
-------------	-----

## 6- GENEL İŞLEM OPERATÖRLERİ

AKTARMA	$=$
PARANTEZ	$()$

## ALGORİTMALARDA ÖZEL TERİMLER

Algoritmalar hazırlanırken sıklıkla kullanılan 3 farklı deyim vardır. Bunlar; değişken, aktarma ve sayaç ifadeleridir.

- 1- DEĞİŞKEN:** Program değişik çalışmalarında, farklı değerler alabilen bilgi alanlarına değişken adı verilir. Değişkenler isimlendirilirken aşağıdaki hususlara mutlaka dikkat edilmelidir.
  - İngiliz alfabesinin A- Z arası büyük ve küçük 26 harfi kullanılabilir.
  - 0 – 9 arası rakamlar ilk karakter olmamak kaydıyla kullanılabilir.
  - Alt çizgi ( \_ ) dışında hiçbir özel karakter değişken isimlendirmelerinde kullanılamaz.
- 2- AKTARMA:** Herhangi bir bilgi alanına veri yazma, herhangi bir ifadenin sonucunu başka bir değişkende gösterme gibi görevlerde “aktarma” işlemi kullanılır.

**Değişken = İfade;**

Yukarıdaki satırda “=” operatörü aktarma operatörü anlamına gelir ve sağdaki ifadenin değerini, soldaki değişkene aktarır. Bu durumda değişkenin daha önceden aldığı değer varsa silinir.

- 3- SAYAÇ:** Programlarda bazı işlemlerin tekrar yapılması veya belirli aralıktaki ardışık değerlerin kullanılması gerekebilir. Bu durumda işlemi uzun uzun yazmak yerine sayaç (Döngü) kullanılabilir. Genel kullanımını aşağıdaki gibidir.

**Döngü Değişkeni = Döngü Değişkeni + (-) Artış miktarı**

**S=S+1**

### DÖNGÜ OLUŞTURMA KURALLARI:

- 1- Döngü değişkenine başlangıç değeri verilir.
- 2- Döngünün artış veya azalış miktarı belirlenir.
- 3- Döngünün bitiş değeri belirlenir.
- 4- Eğer döngü, karar ifadeleriyle oluşturulduysa, sayaç değişkeni, döngü içerisinde adım miktarı kadar arttırılmalı ya da azaltılmalıdır.

Döngüler sıklıkla ardışık toplama veya ardışık çarpma işlemi uygulanır.

## **ARDIŞIK TOPLAMA**

Çalışma prensibi sayaç gibidir. Programlarda, aynı değerin üzerine yeni değerler eklemek için kullanılır. Genel kullanımı;

$$\text{Toplam değışkeni} = \text{Toplam değışkeni} + \text{Sayı}$$

şeklindedir.

Not: Toplam değışkenine başlangıç değeri olarak sıfır (0) atanır.

## **ARDIŞIK ÇARPMA**

Ardışık çarpma işleminde aynı değeri yeni değerlerle çarpılır. Genel kullanımı aşağıdaki gibidir.

$$\text{Çarpım Değışkeni} = \text{Çarpım Değışkeni} * \text{Sayı}$$

Not: Çarpım değışkenine başlangıç değeri olarak bir (1) atanır.

## **KAYNAKLAR:**

- 1- ALGORİTMA GELİŞTİRME VE PROGRAMLAMAYA GİRİŞ, FAHRİ VATANSEVER, SEÇKİN YAYINEVİ**
- 2- VERİ YAPILARI VE ALGORİTMALAR, RİFAT ÇÖLKESEN, PAPTAYA BİLİM YAYINEVİ**
- 3- ALGORİTMA VE PROGRAMLAMA MANTIĞI, BURAK TUNGUT, KODLAB**