

## 5.ÜNİTE: PROBLEM ÇÖZME VE PROGRAMLAMA

### PROBLEM KAVRAMI

Günlük hayatta karşılaştığınız ve çözülmesi gereken durumlara problem denir.

Bir durumun problem olarak tanımlanabilmesi için şu iki şartı sağlamalıdır:

- 1) Can sıkıcı bir durum olmalı.
- 2) Birden fazla çözüm yolu olmalı.

### Günlük Hayatta Karşılaştığımız Problem Örnekleri

- İş kazalarının olması
- Çevre kirliliğinin artması
- Depremde evlerin yıkılması
- Trafik kurallarına uyulmaması
- Sabahları sürekli okula geç kalma
- Derste öğretmen konuşurken dinlememe

**Günlük hayatında karşılaştığınız bir problemi düşünün. Bu problemi nasıl çözebileceğine dair bir öneri yaz. Çözümünü açıklarken:**

**\*Problemin ne olduğunu belirtin:.....**

**\*Bu problemi çözmek için hangi teknolojik araçları veya yöntemleri kullanabileceğinizi yazınız:.....**

### ÖRNEK-1:

**Problemin ne olduğunu belirt:**

*“Ödevlerimi zamanında yapamıyorum ve çok stres oluyorum.”*

**Bu problemi çözmek için hangi teknolojik araçları veya yöntemleri kullanacağını yaz:**

*Takvim Uygulaması: Telefonumda takvim uygulamasına ödevlerin tarihlerini yazabilirim.*

*Zamanlayıcı: Ödev çalışırken bir zamanlayıcı kullanarak belirli sürelerde çalışıp molalar verebilirim.*

*Ödev Listesi: Not defterimde bir yapılacaklar listesi oluşturarak hangi ödevi yapmam gerektiğini görebilirim.*

### ÖRNEK-2:

**Problemin ne olduğunu belirt:**

*“Dışarıda yağın yağmurda eve geç kalıyorum.”*

**Bu problemi çözmek için hangi teknolojik araçları veya yöntemleri kullanacağını yaz:**

*Hava Durumu Uygulaması: Telefonumda hava durumu uygulamasını kontrol ederek yağmurun ne zaman başlayacağını öğrenebilirim.*

*Şemsiye: Yanımda her zaman bir şemsiye taşıyarak yağmurdan korunabilirim.*

*Planlama: Yağmur yağacağını öğrendiğimde, daha erken çıkmayı planlayabilirim.*

### Problem Çözme Adımları:

- 1-Problem Anlama
- 2-Plan Yapma
- 3-Planı Uygulama
- 4-Çözümü Değerlendirme

### Problem Çözme Yöntem ve Teknikleri

- Beyin Fırtınası
- Deneme Yanılma
- Zihin Haritası Çıkarma
- Takım Çalışması

### Problem Çözme Stratejileri

- Deneme Yanılma
- Tahmin Yürütme
- Örüntüleri Kontrol Etme
- Tersten Gitme
- Mantık Yürütme
- Görselleştirme

**Örnek:** Bir kapının önüne geldiniz. O kapıyı açıp geçmek istiyorsunuz fakat kilitli olduğu için açamadınız. Duvarda bir miktar anahtarın asılı olduğunu fark ettiniz. Kapıyı açmak için girişimde bulunmak istediniz. Bu senaryoda uygulananlar problem çözme stratejilerinden **DENEME YANILMA**'ya örnek sayılabilir.

## Sabit ve Değişkenler

\* Asla değişmeyen ifadelere **SABİT** denir.

\* Farklı değerler alabilen ifadelere **DEĞİŞKEN** denir.

Bir marketin bulunduğu konum ve marketteki ürün rafları sabit, marketteki ürünler ve alışveriş için gelen müşteriler ise değişkendir.

## Operatör, İfade ve Eşitlikler

Bir aracı, nesneyi ya da sayıyı işletmek/çalıştırmak anlamında kullanılan ifadeye **operatör** denir.

## Aritmetiksel Operatörler

5+9=14 11x12=132

Matematikte kullanılan +, -, x, /, = gibi operatörlere **ARİTMETİKSEL OPERATÖRLER** denir.

## Mantıksal Operatörler ve Eşitlikler

Günlük hayatta karşılaştığımız problemlerde herhangi bir tercih, karşılaştırma, eşitlik gibi seçeneklerle karşılaşabiliriz. Bir bilgisayar programı oluştururken bazı terimler kullanılır. Bunlar mantıksal operatör veya eşitliklerdir. İşte bu mantıksal operatör veya eşitlikler: **VE, VEYA, DEĞİL, KÜÇÜKTÜR, BÜYÜKTÜR, KÜÇÜK EŞİTTİR, BÜYÜK EŞİTTİR** dir.

## ALGORİTMA

Bir problemi çözmek veya bir görevi tamamlamak için adım adım listelenmiş talimatlara **algoritma** denir.

## Örnek: Ayran Yapma Algoritması

ADIM 1-Başla

ADIM 2-Yoğurdu kaba koy

ADIM 3-Su ekle

ADIM 4-Tuz ekle

ADIM 5-Çırp

ADIM 6-Bardağa doldur

ADIM 7-Bitir

## Algoritmik Düşünmenin Faydaları

Bir problemi çözmek için gerekli olan talimatları doğru sıralamada yaptığınızda problemi parçalara ayırmış olursunuz. Böylece her bir adımı kontrol edebilir, varsa hatanızı kolaylıkla tespit edebilirsiniz. Özetle bir problemi çok daha basit parçalara ayırarak daha kolay anlaşılmasını ve çözülmesini sağlarız.

Algoritmik düşünmek, hata yapma riskini azaltır. Problem çözme yeteneğinizi artırır. Ayrıca algoritmik düşünce, bilgisayar biliminin anahtar parçalarından biridir. Programlama yapmayı düşünenler öncelikle algoritma yazar. Bir programcı ortaya çıkaracağı yazılım öncesinde, yapacağı işlemleri belirten adımları ve kuralları içeren algoritma tasarlar.

## AKIŞ ŞEMASI

Bir sürecin adımlarını geometrik şekillerle gösteren çizime akış şeması denir. Bir algoritmayı görsel olarak daha kolay takip etmek için akış şeması kullanılır. Şemadaki her şeklin bir kullanım amacı vardır.

## Akış Şeması Sembolleri ve Anlamları

<b>Başla</b> ve <b>Bitir</b> adımları için kullanılır. Akış şemasının başlangıç ve bitiş kısmında kullanılır.	<b>Giriş</b> ya da <b>Çıkış</b> işlemleri için kullanılır. Örneğin klavyeden bir sayı girilmesinin istenmesi.	<b>Hesaplama</b> ya da <b>Değişkene Değer Atama</b> işlemleri için kullanılır. Örneğin; iki sayıyı topla veya girilen sayıyı A değişkenine ata.	<b>Karşılaştırma</b> ya da <b>Karar Verme</b> işlemleri için kullanılır. Örneğin; girilen sayının negatif olup olmadığına karar verme.	Sonucu <b>ekrana yazdırmak</b> için kullanılır. Örneğin; Ekran "geçti" yaz.	Talimatların yürütülme sırasını gösterir. Aynı zamanda diğer şekiller arasındaki ilişkiyi temsil eder.