

2024-2025 EĞİTİM VE ÖĞRETİM YILI

PROGRAMLAMAYA GİRİŞ VE ALGORİTMA DERSİ

3. ÜNİTE

PROGRAMLAMAMANIN TEMEL KAVRAMLARI

Programlama Temelleri

-
- Programlamada sıkça kullanılan temel kavramlar:
 - - Operatörler
 - - Değişkenler
 - - Koşullar
 - - Döngüler
 - Örnek: `x = 5`, `if`, `while`

Operatörler ve Temel İşlemler

- Programlamada matematiksel işlemler (toplama, çıkarma, çarpma, bölme)
- - Karşılaştırma operatörleri: ==, !=, <, >, <=, >=
- Örnek: `a == b`, `a != b`, `a > b`

Koşullar ve Döngüler

Koşul yapıları: if-else,
döngüler: for ve while
döngüleri.

- Örnek: Sayının pozitif olup
olmadığını kontrol eden if
yapısı.

- Döngü örneği: 1'den 10'a
kadar sayıları yazdıran for
döngüsü.

Matematiksel ve Programlama Sembolleri Arasındaki Farklar

Matematikte '=' sembolü eşitlik anlamına gelirken, programlamada atama anlamında kullanılır.

- Örnek: $x = 5$ ifadesi, 'x'i 5'e eşitle' anlamına gelir.

- Ek olarak '==' karşılaştırma operatörü örneği.

Sembol Farklılıkları Örnekleri

Farklı programlama dillerinde aynı sembolün farklı anlamlar ifade etmesi.

- Örnekler: '==' operatörü Python ve JavaScript'te aynı ama '===' yalnızca JavaScript'te mevcut.

Pratik Uygulamalar ve Örnek Sorular

Kullanıcıdan bir sayı alıp pozitif mi, negatif mi olduğunu söyleyen bir program yazın.

Veri Kavramı ve Veri Türlerinin Tanımı

- Veri nedir? Farklı veri türleri neden vardır?
- - Sayısal veriler (5, 3.14), metinsel veriler ('Merhaba').

Veri Türlerini Doğru Tanımlamanın Programlamadaki Önemi

- Doğru veri türünün seçilmesi neden önemlidir?
- - Bellek yönetimi ve işlemlerin hızlandırılması için gereklidir.

Farklı Programlama Dillerinde Veri Türleri

Her programlama dilinin kendi veri türü yapısı vardır.

- Örnekler:
Python'da int, float, str; Java'da int, double, String.

Verinin Kullanım Amacına Uygun Saklanması

Verinin programda hangi amaçla kullanılacağını belirlemek önemlidir.

- Aynı veri farklı durumlarda farklı veri türü olarak tanımlanabilir.

- Örnek: 5 sayısı yaş olarak tamsayı (integer), fiyat olarak ondalıklı sayı (float).

Veri Türlerinin İsimlendirmesi Diller Arasında Değişebilir

Örneğin,
Python'da 'str'
olarak bilinen
veri türü C++
dilinde 'string'
olarak kullanılır.

Sayısal ve Metinsel Veri Türleri Örnekleri

Sayısal veri türleri:
Tamsayı (int),
ondalıklı sayı (float).

Metinsel veri
türleri: Metin (str
veya string).

- Örnekler: int x =
10, float y = 3.14,
string isim = 'Ali'.

Veri Türlerinin Program Amacına Göre Kullanımı

Programın amacına göre veri türlerinin farklılıkları

- Sayı, metin, dizi gibi veri türlerinin programın amacı doğrultusunda tanımlanması gerekir.

Örnek: Yaş, maaş veya kullanıcı adı gibi farklı verilerin türlerine göre tanımlanması.

Veri Türlerine Göre Alt Türler

- Seçilen veri türüne göre alt veri türleri Örnekler: Tam sayılar (int), ondalıklı sayılar (float), uzun tamsayılar (long).

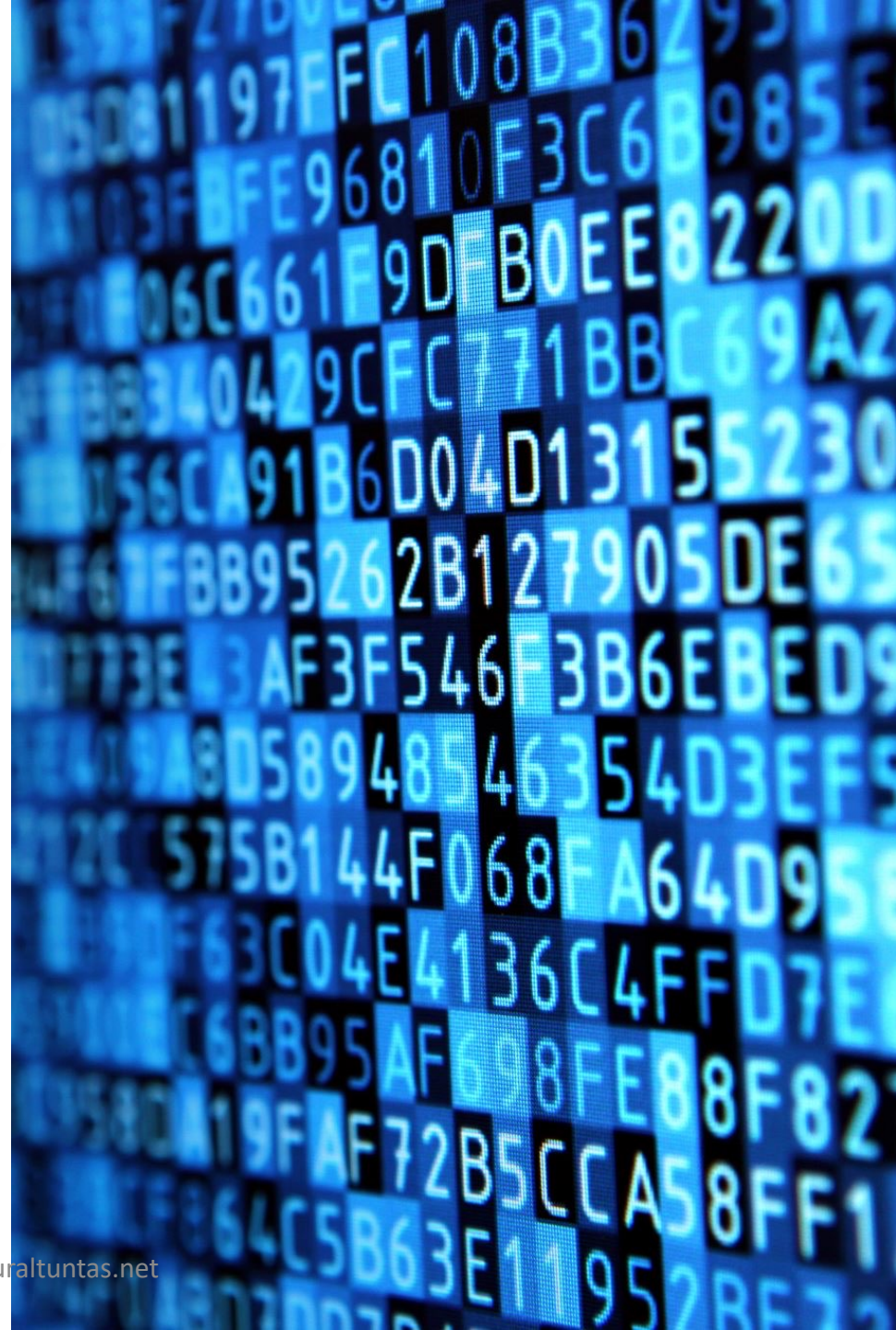


Sayısal Veri Türleri: Tamsayı ve Ondalıklı Sayı Kullanımı

- Tamsayı (int) ve ondalıklı sayı (float) veri türleri arasındaki farklar.
- - Örnek: Yaş gibi tam sayılar int ile, fiyat gibi ondalıklı sayılar float ile tanımlanır.
- - Bu tanımların program içindeki sonuçları nasıl etkileyebileceğine dair örnekler.

Byte, Integer, Long, Float, String, Boolean, Char Kavramları

- Her veri türü tanımlanır ve örneklerle açıklanır:
- - byte: 0-255 arası küçük sayılar için.
- - int: Tamsayılar.
- - long: Büyük tamsayılar.
- - float: Ondalıklı sayılar.
- - string: Metinler.
- - boolean: True/False değerleri.
- - char: Tek karakterler.



Dizi (Array) Kavramı

Aynı türde birden fazla veriyi depolamak için diziler kullanılır.

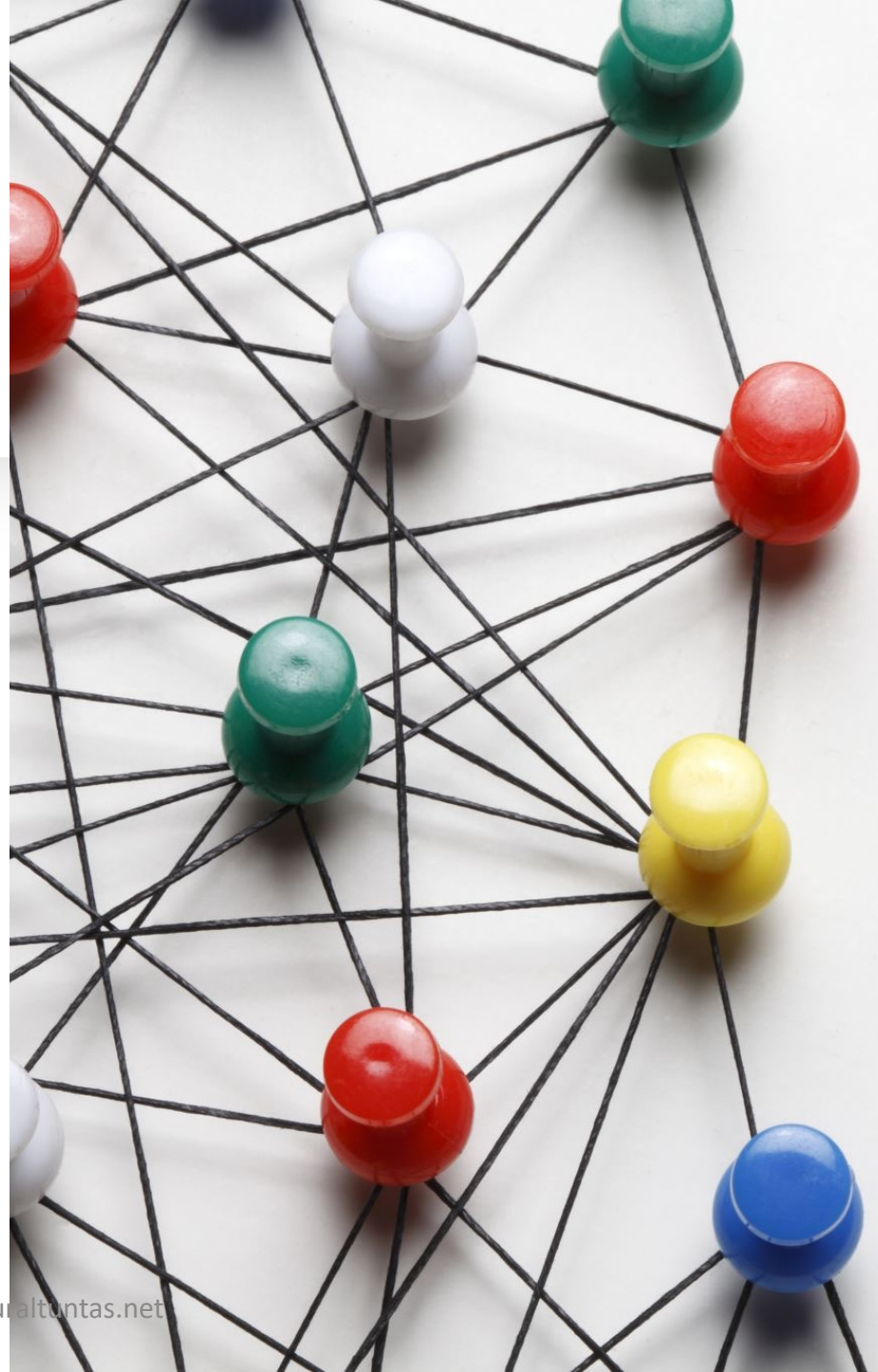
- Örnek: Öğrencilerin yaşlarını veya isimlerini depolayan diziler (`int[]` yaslar, `string[]` isimler).

[illegible]

- Kaynak: onur

Mantıksal Operatörler

- Karar yapılarında mantıksal operatörlerin rolü: `&&` (ve), `||` (veya), `!` (değil).
- - Örnek: `age >= 18 && citizen == true` ile çoklu koşul oluşturma.



If-Else ve Else-If Yapıları

- Program akışında if-else ve else if yapılarına örnekler.
- - Örnek:
 - `if (age >= 18) {...}`
 - `else if (age < 18) {...}`
 - `else {...}`



Birden Fazla Koşul ile Program Tasarlama

- Örnek: Bir öğrencinin notuna göre “Geçti”, “Kaldı” veya “Teşekkür” mesajı veren program.



Döngü Yapılarının Programlara Sağladığı Kolaylıklar

Programda tekrar eden işlemleri kullanmanın faydaları.



- Döngüler, programcılarının iş yükünü azaltarak kodu daha verimli hale getirir.



Kod Tekrarının Önüne Geçmek ve Anlaşılır Programlar Yazmak

Döngüler sayesinde kod tekrarını önleyerek daha sade ve anlaşılır programlar yazmak mümkündür.

- Örnek: 1'den 10'a kadar sayıları yazdırmak için döngü kullanmak.

- Döngü Yapıları:
For ve While Döngüleri

Programın amacına göre hangi döngü yapısının seçileceği önemlidir.

- For ve while döngülerinin işlevi ve kullanım alanları.

Döngülerin Temel Taşları

Başlangıç ve bitiş değeri, artış miktarı, devam etme koşulu gibi döngü elemanları açıklanır.

- Örnek:

```
for (int i = 0; i < 10; i++)
```

Döngü ile ve Döngüsüz Problem Çözümü

Aynı problemin
döngü ile ve döngü
olmadan çözülmesi
örneklenir.

- Örnek: Ekrana 5 kez
'Merhaba' yazdıran
program ile kod
farkının gösterilmesi.

Döngü ile ve Döngüsüz Problem Çözümü

Problem:

Bir markette indirim kampanyası yapılmaktadır. Kullanıcıya sepetinde bulunan ürünlerin sayısı ve her bir ürünün fiyatı sorulacaktır. Kullanıcı her bir ürünün fiyatını girdikten sonra, toplam fiyat hesaplanacaktır. Eğer toplam fiyat 100 TL'nin üzerinde ise, kullanıcıya %10 indirim uygulanacaktır. Son olarak, indirimin uygulandığı ya da uygulanmadığı toplam fiyat ekrana yazdırılacaktır.

Döngü ile ve Döngüsüz Problem Çözümü

Çözüm Adımları ve Algoritma:

1.Başla

2.Kullanıcıdan ürün sayısını al.

3.ToplamFiyat adlı bir değişken oluştur ve 0 olarak başlat.

4.Ürün sayısı kadar döngü oluştur:

1. Her döngüde kullanıcıdan ürünün fiyatını al ve ToplamFiyata ekle.

5.Eğer ToplamFiyat 100 TL'den büyükse:

1. ToplamFiyata %10 indirim uygula.

6.ToplamFiyatı ekrana yazdır.

7.Bitir

Döngü ile ve Döngüsüz Problem Çözümü

Akış diyagramını çizin

•Başlangıç

- ÜrünSayısı sorulur, alınır.

- ToplamFiyat = 0

- Döngü başlatılır (i = 1'den ÜrünSayısı'na kadar):

- Her döngüde ÜrünFiyatı alınır.

- ToplamFiyat = ToplamFiyat + ÜrünFiyatı

- Koşul: ToplamFiyat > 100?Evet: ToplamFiyat = ToplamFiyat * 0.9

- Hayır: Hiçbir işlem yapılmaz.

ToplamFiyat yazdırılır.

Son