



**NİŞANTAŞI  
ÜNİVERSİTESİ**

**İSTANBUL NİŞANTAŞI ÜNİVERSİTESİ  
UYGULAMALI BİLİMLER YÜKSEKOKULU  
VERİ BİLİMİ VE ANALİTİĞİ BÖLÜMÜ**

**Fatih OKUMUŞ**  
**Algoritma ve Programlama Dönem İçi Uygulama ve Gelişim**

**Yüksekokulu Projesi**

**İSTANBUL  
Aralık 2025**

# İÇİNDEKİLER

Sunum ve Dosya İçeriğinin Özeti .....	0
Temel Algoritma Örnek Sorularının Çözümü .....	1
Program Dilleri ve A* (A Star) Teoremi İncelemesi .....	2
<b>Manhattan Mesafesi</b> .....	3
<b>Avantajları</b> .....	3
• Dijkstra'dan daha hızlıdır, çünkü tahmin kullanır. ....	3
<b>Dezavantajları</b> .....	3
• Hesaplama yükü fazladır (büyük haritalarda yavaşlayabilir) .....	3
<b>Nasıl Çalışır?</b> .....	3
• A* algoritması her adımda bir düğüm (node) seçer ve o düğümün toplam “maliyetini” hesaplar: .....	3
Algoritma Sorularının Algoritmik Doğal Dil Çözümlemesi .....	5
Algoritma Sorularının Akış Diyagramı İle Çözümlemesi .....	6
Dönem Çalışmalarının Genel Değerlendirilmesi .....	17

## Sunum ve Dosya İeriğinin zeti

Bilgi teknolojilerindeki hızlı gelişim, algoritmik düşünme ve programlama becerilerini günümüzün temel gereksinimleri arasına taşımıştır. Özellikle veri bilimi alanında, karmaşık problemlerin sistematik ve doğru bir şekilde çözümlenmesi büyük önem taşımaktadır. Bu problemlerin çözümünde algoritmalar, izlenecek adımların belirlenmesini sağlarken; programlama dilleri, bu adımların bilgisayar ortamında uygulanmasına olanak tanımaktadır.

Bu çalışma, İstanbul Nişantaşı Üniversitesi Uygulamalı Bilimler Yüksekokulu Veri Bilimi ve Analitiğı Bölümü kapsamında yürütölen Algoritma ve Programlama dersi çerçevesinde hazırlanmıştır. Çalışmada; algoritma örnek soruları, programlama dilleri ve A\* algoritması, algoritmik doğal dil ile problem çözme yöntemleri ve akış diyagramları kullanılarak algoritma tasarımı konuları ele alınmıştır. Bu doğrultuda amaç, öğrencinin algoritmik düşünme yeteneğini geliştirmek, problem çözme sürecini yapılandırmak ve veri bilimi alanında kullanılan temel programlama altyapısını sağlamlaştırmaktır.

## Temel Algoritma Örnek Sorularının Çözümü

### Soru 1

- Öğrencinin 3 adet sınav notu bulunuyor. Bu notların ortalamasına göre öğrencinin geçti veya kaldı sonucunu veren bir algoritma tasarlayınız.

### Soru 2

- Alınan bir sayının çift mi yoksa tek mi olduğunu belirleyen bir algoritma tasarlayınız.

# Program Dilleri ve A\* (A Star) Teoremi İncelemesi

## Program Dilleri Nelerdir?

- Pascal, Basic, C, C#, C++, Java, JavaScript, Cobol, Perl, PHP, Python, Ada, Fortran, Delphi ve Swift

## Python nedir ne işe yarar?

**Python;** web uygulamaları, yazılım geliştirme, veri bilimi ve makine öğreniminde (ML) yaygın olarak kullanılan bir programlama dilidir.

## Soru 1

- Bir kişinin anksiyete (kaygı) düzeyini ölçmek için 5 soruluk bir test yapılmaktadır. Her soruya 1 ile 5 arasında puan verilir:

1 = Hiç katılmıyorum  
5 = Tamamen katılıyorum

Toplam puana göre anksiyete düzeyi belirlenir:

- 5–10 Arası: Düşük
- 11–20 Arası: Orta
- 21–25 Arası: Yüksek

1. Adım: Başla.

2. Adım: Kullanıcıdan 5 sorunun her biri için puan iste.

3. Adım: Her puanı ayrı değişkenlerde tut (örneğin: p1, p2, p3, p4, p5).

4. Adım: Toplam puanı hesapla:  $\rightarrow \text{toplam} = (p1 + p2 + p3 + p4 + p5)$ .

5. Adım: Eğer toplam  $\geq 5$  ve  $\leq 10$  ise  $\rightarrow$  “Anksiyete düzeyi: Düşük” yazdır.

6. Adım: Değilse eğer toplam  $\geq 11$  ve  $\leq 20$  ise  $\rightarrow$  “Anksiyete düzeyi: Orta” yazdır.

7. Adım: Aksi halde,  $\rightarrow$  “Anksiyete düzeyi: Yüksek” yazdır.

8. Adım: Toplam puanı ve sonucu ekrana göster.

9. Adım: Dur.

## A\* algoritması Ne işe Yarar?

A\* algoritması, en kısa ve en verimli yolu bulmak için kullanılan bir arama (pathfinding) algoritmasıdır.

Bir noktadan (başlangıç) diğerine (hedef) giderken:

- Mesafenin en kısa,
- Geçişin en az maliyetli ve en mantıklı rotayı bulur.

Örneğin:

- Navigasyon uygulamaları (en kısa yol bulma)
- Oyunlarda karakterin hedefe gitmesi
- Robotların yol planlaması gibi durumlarda kullanılır.

## Manhattan Mesafesi

- Mevcut hücrenin x ve y koordinatları ile hedef hücrenin x ve y koordinatları arasındaki farkların mutlak değerlerinin toplamıdır.

$$h = \text{abs}(\text{düğüm.x} - \text{hedef.x}) + \text{abs}(\text{düğüm.y} - \text{hedef.y})$$

- Bu buluşsal yöntem, 2 boyutlu bir ızgarada yalnızca 4 yönde (yukarı, aşağı, sol, sağ) hareket edebildiğimiz durumlarda kullanılabilir.

## Avantajları

- Dijkstra'dan daha hızlıdır, çünkü tahmin kullanır.
- En kısa yolu bulmada kesin ve verimlidir.

## Dezavantajları

- Hesaplama yükü fazladır (büyük haritalarda yavaşlayabilir).
- Tahmin fonksiyonu (heuristic) doğru seçilmezse yanlış sonuç verebilir.

## Nasıl Çalışır?

- A\* algoritması her adımda bir düğüm (node) seçer ve o düğümün toplam “maliyetini” hesaplar:

$$f(n)=g(n)+h(n) \quad f(n) = g(n) + h(n) \quad f(n)=g(n)+h(n)$$

- $g(n)$ : Başlangıç noktasından şu ana kadar olan gerçek maliyet
- $h(n)$ : Hedefe kalan tahmini maliyet (heuristic)

- $f(n)$ : Toplam tahmini maliyet
- Algoritma, en küçük  $f(n)$  değerine sahip yolu seçerek hedefe ulaşır.

**Tablo 1. Benzer Algoritmaların Karşılaştırma Tablosu**

Algoritma	Tahmin (Heuristic)	En kısa yolu bulur mu?	Hız	Bellek kullanımı
A*	Var	Evet	Hızlı	Orta
Dijkstra	Yok	Evet	Yavaş	Orta
GreedyBFS	Var	Hayır	Hızlı	Düşük
IDA*	Var	Evet	Orta	Az
BFS	Yok	Evet	Yavaş	Çok
DFS	Yok	Hayır	Hızlı	Az
JPS	Var	Evet	ÇokHızlı	Orta

**Kaynak:** Russell, S., & Norvig, P. (2021). *Artificial Intelligence: A Modern Approach* (4th ed.). Pearson. (A\*, Greedy Best-First Search, IDA\*)

## Algoritma Sorularının Algoritmik Doğal Dil Çözümlemesi

### Soru 1

- Öğrencinin 3 adet sınav notu bulunuyor. Bu notların ortalamasına göre öğrencinin geçtiği veya kaldı sonucunu veren bir algoritma tasarlayınız.

Adım No	İşlem
1	Başla
2	Birinci sınav notunu al $\rightarrow$ not1
3	İkinci sınav notunu al $\rightarrow$ not2
4	Üçüncü sınav notunu al $\rightarrow$ not3
5	Ortalamayı hesapla: $\text{Ortalama} = (\text{not1} + \text{not2} + \text{not3}) / 3$
6	Eğer $\text{Ortalama} \geq 50$ ise "Geçti" yaz. Aksi halde "Kaldı" yaz.
7	Bitir

### Soru 2

Alınan bir sayının çift mi yoksa tek mi olduğunu belirleyen bir algoritma tasarlayınız.

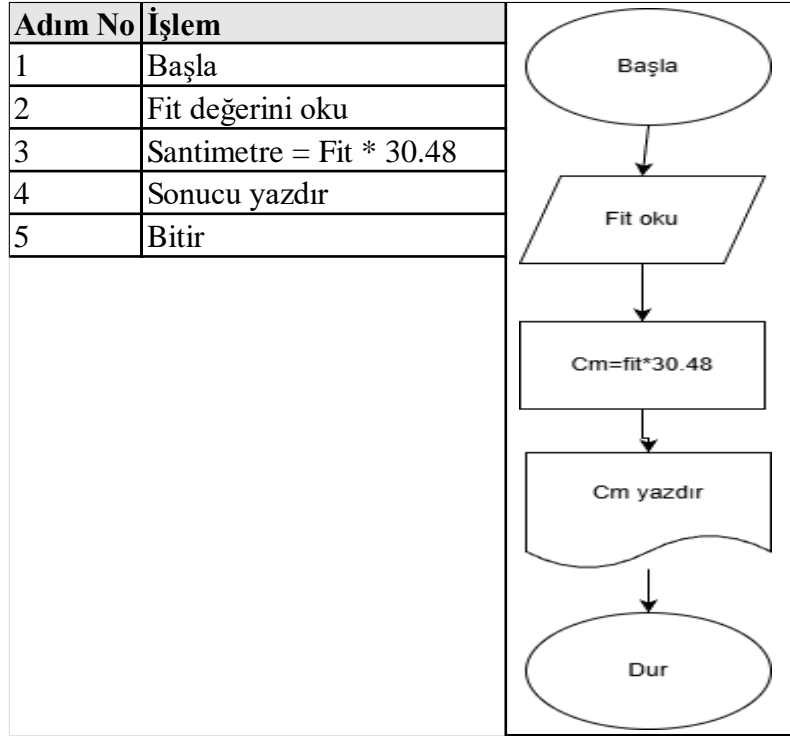
Adım No	İşlem
1	Başla
2	Bir sayı al $\rightarrow$ sayi
3	sayi mod 2 işlemini yap (yani sayıyı 2'ye böl, kalanı bul)
4	Eğer kalan = 0 ise $\rightarrow$ "Sayı çifttir" yaz. Aksi halde $\rightarrow$ "Sayı tektir" yaz.
5	Eğer $\text{Ortalama} \geq 50$ ise $\rightarrow$ "Geçti" yaz. Aksi halde $\rightarrow$ "Kaldı" yaz.
6	Bitir



## Algoritma Sorularının Akış Diyagramı İle Çözümlemesi

### Soru 1

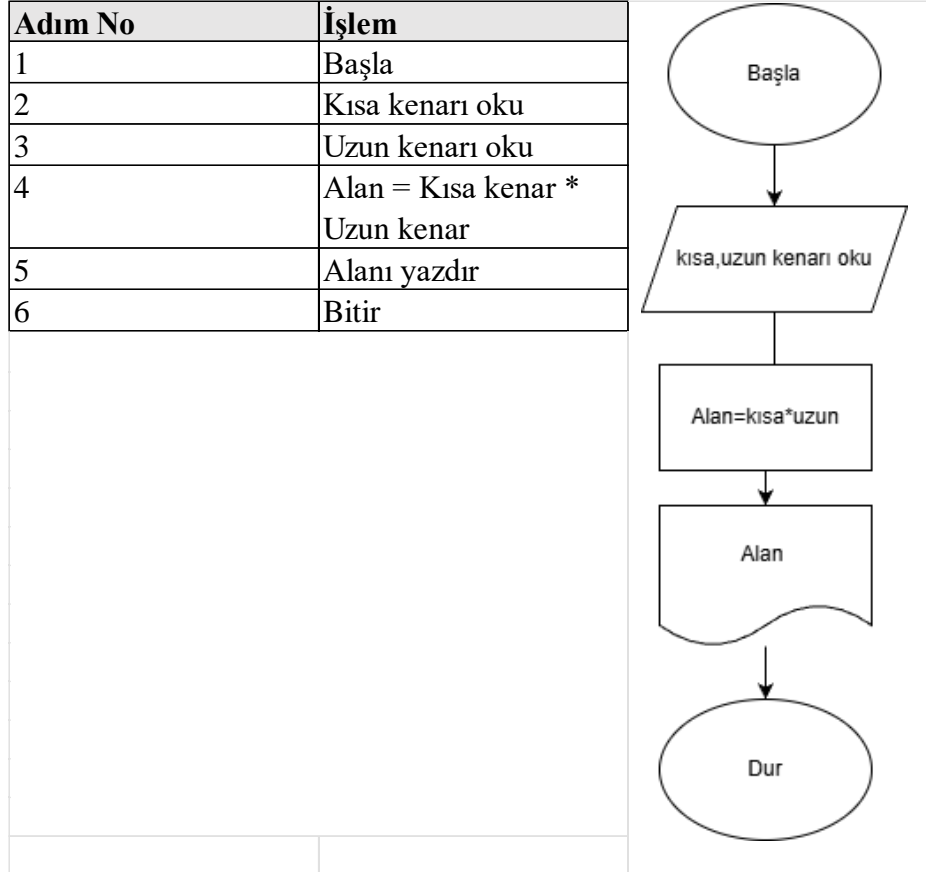
Uzunluğu fit cinsinden santimetreye dönüştürmek için bir akış şeması çizin.



1 fit = 30.48 cm dönüşüm oranı kullanılır.

## Soru 2

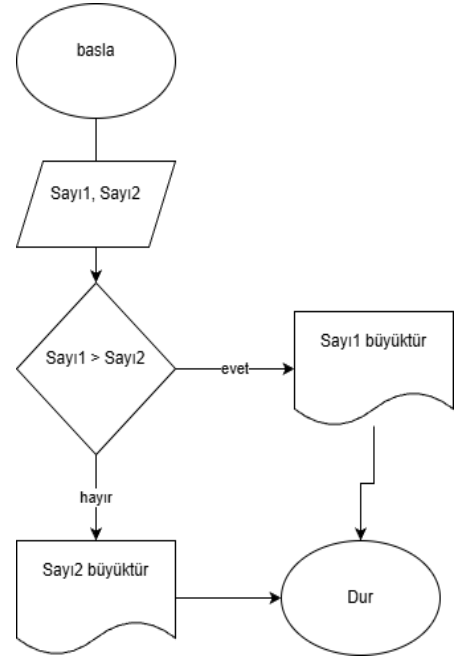
Bir dikdörtgenin iki kenarını okumak ve alanını hesaplamak için bir akış şeması çizin.



### Soru 3

İki değeri okuyan, en büyük değeri belirleyen ve en büyük değeri tanımlayıcı bir mesajla yazdırılan bir akış şeması çizin.

Adım No	İşlem
1	Başla
2	Sayı1, Sayı2 oku
3	İkinci sınav notunu al → not2
4	Eğer Sayı1 > Sayı2 ise 'Sayı1 büyüktür' yaz
5	Ortalamayı hesapla: $\text{Ortalama} = (\text{not1} + \text{not2} + \text{not3}) / 3$
6	Değilse 'Sayı2 büyür' yaz
7	Bitir



#### Soru 4

$2^4$ 'ü hesaplamak için bir algoritma yazın ve akış şeması çizin.

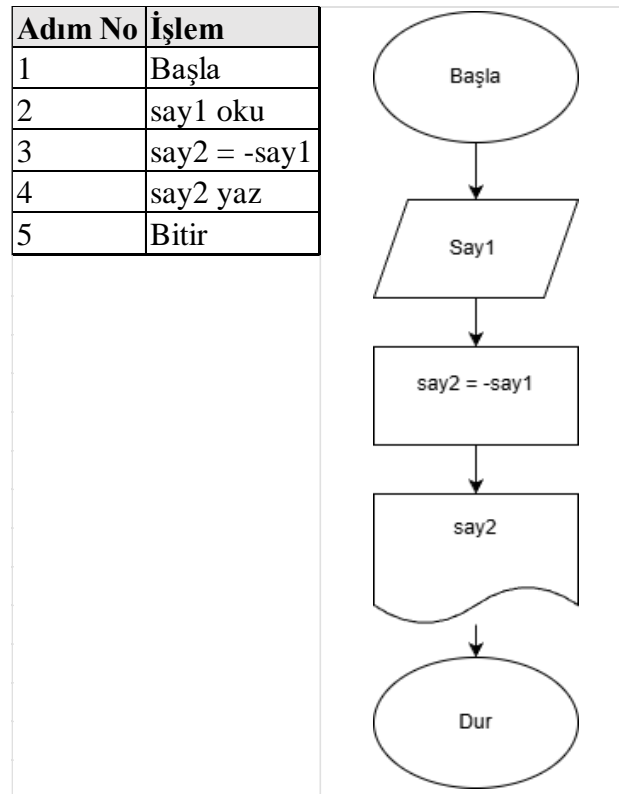
Yöntem 1:  $2^4 = 2 * 2 * 2 * 2$

Yöntem 2 (döngüyle):

Başla  $\rightarrow$  Sonuç=1  $\rightarrow$  i=1  $\rightarrow$  i<=4 ise Sonuç=Sonuç\*2  $\rightarrow$  i=i+1  $\rightarrow$  Döngü bitince Sonuç yaz  $\rightarrow$  Bitir

#### Soru 5

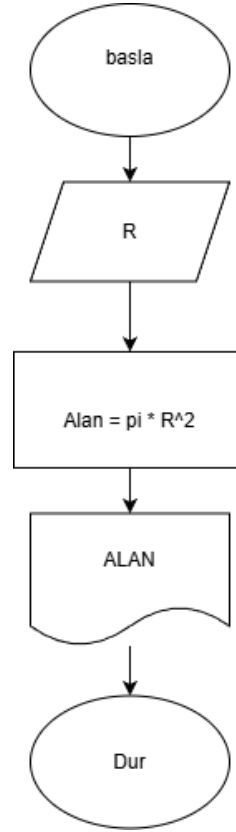
say1 = A değişkenini okumak ve say1'in negatif değerini başka bir say2 değişkeninde depolamak için bir akış şeması çizin.



### Soru 6

Yarıçapı R olan bir dairenin alanını hesaplamak için bir akış şeması çizin.

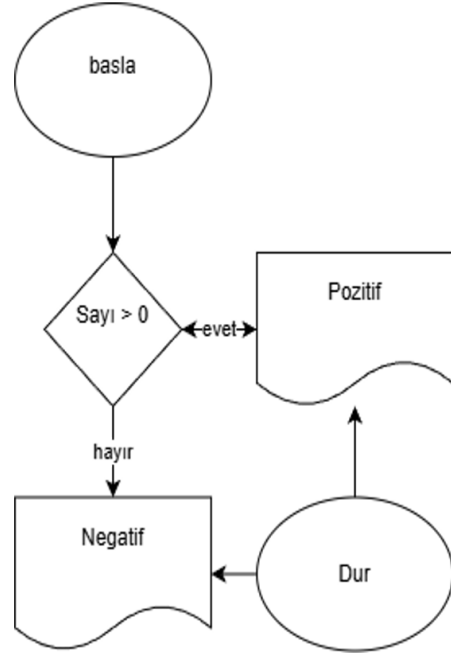
Adım No	İşlem
1	Başla
2	R oku
3	$\text{Alan} = \pi * R * R$
4	Alanı yaz
5	Bitir



### Soru 7

Bir sayıyı okumak ve sayının pozitif mi negatif mi olduğunu belirlemek için bir akış şeması çizin.

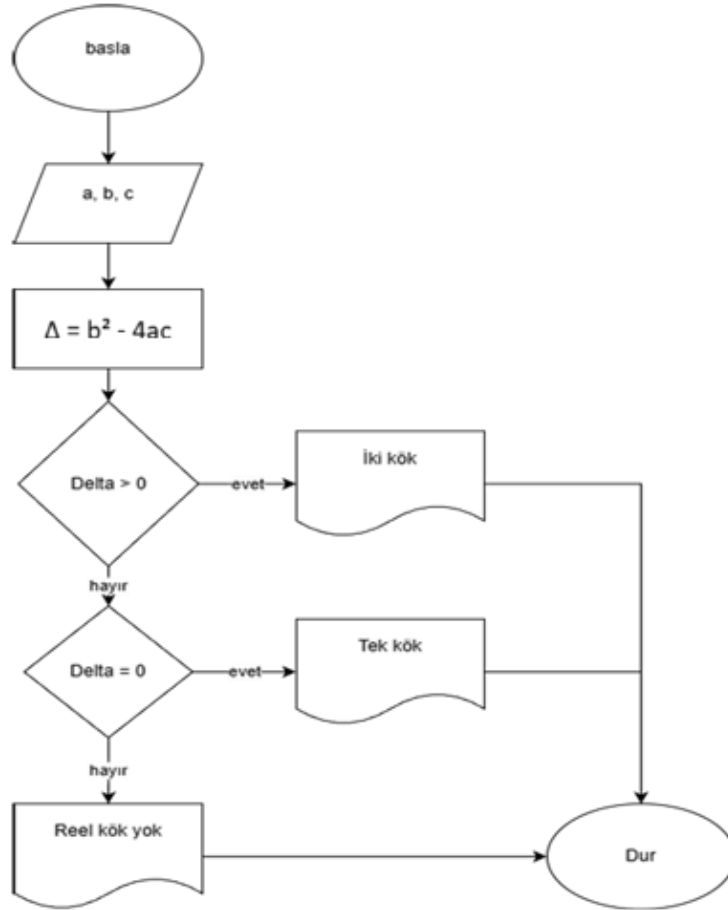
Adım No	İşlem
1	Başla
2	Sayı oku
3	Eğer $\text{sayı} > 0$ ise 'Pozitif' yaz
4	Eğer $\text{sayı} < 0$ ise 'Negatif' yaz
5	Aksi halde 'Sıfır' yaz
6	Bitir



### Soru 8

$ax^2+bx+c=0$  ikinci dereceden denklemin tüm köklerini bulmak için bir akış şeması çizin.

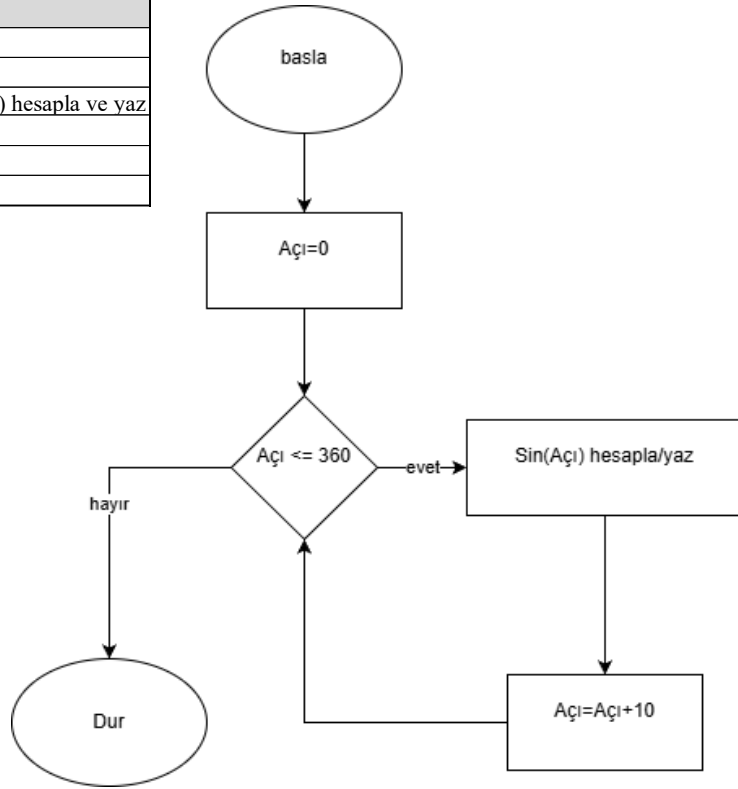
Adım No	İşlem
1	Başla
2	a, b, c oku
3	$\Delta = b^2 - 4ac$
4	Eğer $\Delta > 0$ iki kök
5	$\Delta = 0$ tek kök
6	$\Delta < 0$ reel kök yok
7	Bitir



### Soru 9

0° ile 360° arasındaki açılarının sinüs değerlerini 10°'lik adımlarla elde etmek için bir akış şeması çizin.

Adım No	İşlem
1	Başla
2	$Açı=0$
3	$Açı \leq 360$ ise $\sin(Açı)$ hesapla ve yaz
4	$Açı = Açı + 10$
5	Döngü bitince Bitir
7	Bitir

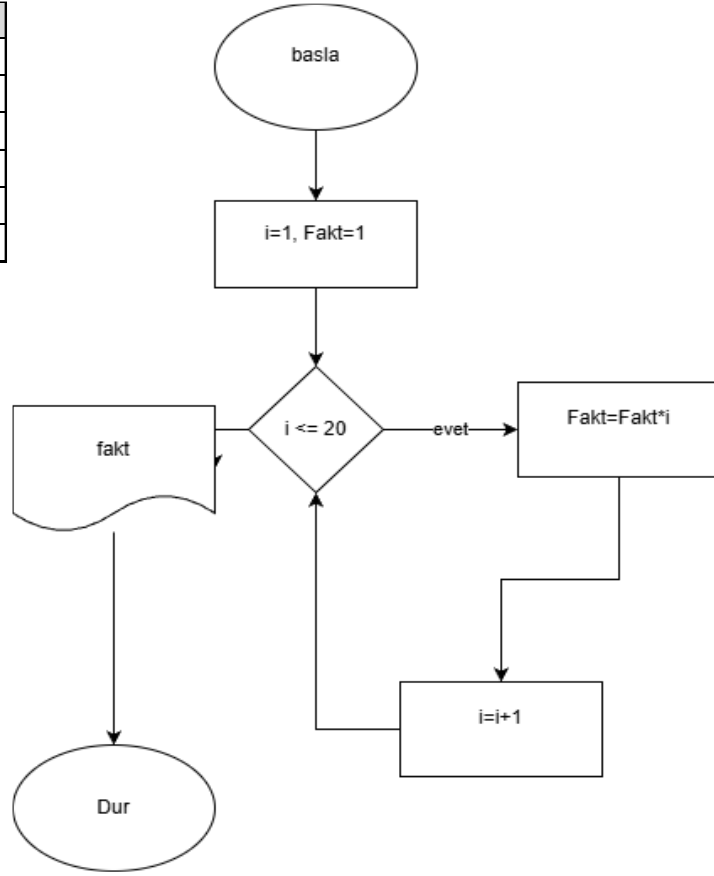




### Soru 10

20'nin faktöriyelini elde etmek için bir akış şeması çizin.

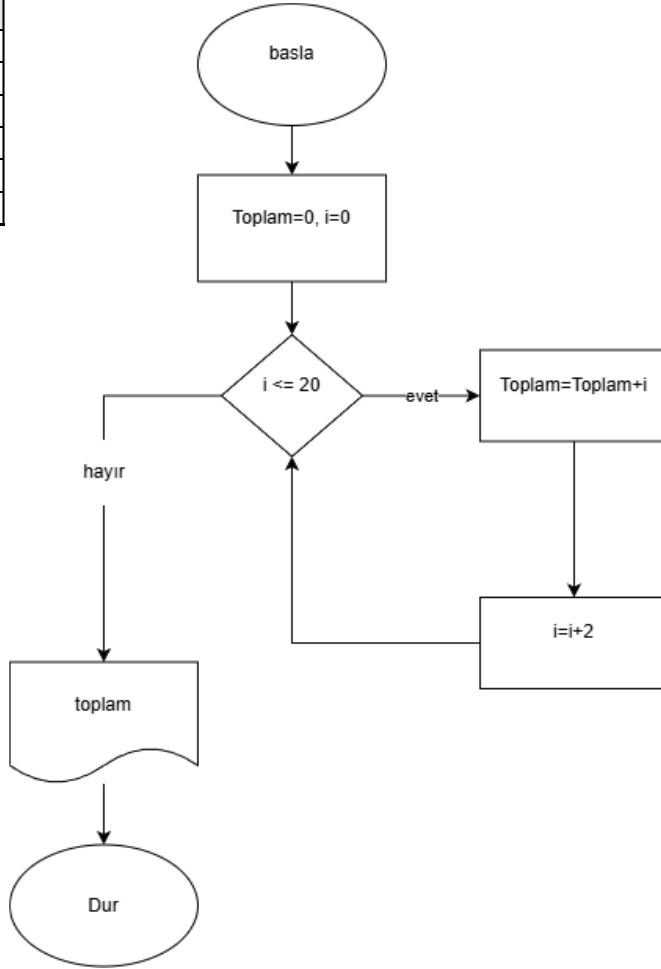
Adım No	İşlem
1	Başla
2	$i=1$ , Fakt=1
3	$i \leq 20$ ise Fakt=Fakt*i
4	$i=i+1$
5	Döngü bitince Fakt yaz
7	Bitir



### Soru 11

0'dan 20'ye kadar çift sayıları eklemek için bir akış şeması çizin.

Adım No	İşlem
1	Başla
2	Toplam=0, i=0
3	$i \leq 20$ ise Toplam=Toplam+i
4	$i=i+2$
5	Döngü bitince Toplam yaz
7	Bitir



**Soru 12**

Bir sayı listesindeki en büyük sayıyı bulmak için bir akış şeması çizin.

Cevap:

Başla → İlk sayıyı EnBüyük olarak al → Her yeni sayı için Eğer sayı > EnBüyük ise EnBüyük = sayı → Tüm sayılar bitince EnBüyük yaz → Bitir

**Soru 13**

N öğrencinin adını ve her öğrenci için üç notu girmek için bir akış şeması çizin ve öğrenci adını, ortalama notunu ve notunu yazdırın.

Cevap:

Başla → N oku → i=1 → i≤N iken: Ad ve 3 not oku → Ortalama=(n1+n2+n3)/3 → Notu belirle → Ad, Ortalama, Not yaz → i=i+1 → Döngü bitince Bitir

**Soru 14**

N sayısını kabul eden ve negatif sayıların toplamını, pozitif sayıların toplamını ve her gruptaki sayıyı bulmak için bir akış şeması çizin.

Cevap:

Başla → N oku → i=1, TopPoz=0, TopNeg=0 → i≤N iken: Sayı oku → Eğer Sayı>0 TopPoz+=Sayı → Değilse TopNeg+=Sayı → i=i+1 → Döngü bitince yaz → Bitir

## Dönem Çalışmalarının Genel Değerlendirilmesi

Bu çalışma kapsamında, Algoritma ve Programlama dersi doğrultusunda algoritmik düşünmenin temel ilkeleri ve problem çözme yaklaşımları ele alınmıştır. Algoritma örnek soruları, programlama dilleri, A\* algoritması, algoritmik doğal dil kullanımı ve akış diyagramları aracılığıyla farklı problem türleri analiz edilmiş ve çözüm süreçleri sistematik bir yapıya kavuşturulmuştur. Bu sayede problemlerin yalnızca sonuç odaklı değil, süreç odaklı bir yaklaşımla ele alınmasının önemi ortaya konulmuştur.

Çalışma sonucunda elde edilen kazanımlar şu şekilde özetlenebilir: algoritmik düşünme becerisinin geliştirilmesi, problemlerin adım adım ve mantıksal bir sırayla çözülmesi, algoritma tasarlama ve akış diyagramı oluşturma yeteneğinin kazanılması, programlama dillerinin kullanım alanlarının kavranması ve A\* algoritması gibi arama algoritmalarının çalışma mantığının öğrenilmesi. Ayrıca, doğal dil kullanılarak algoritma ifade edebilme ve bu ifadeleri görsel yapılarla destekleyebilme becerileri geliştirilmiştir. Bu kazanımlar, öğrencinin veri bilimi ve analitiği alanında karşılaşacağı problemlere daha analitik, sistematik ve bilinçli bir yaklaşım geliştirmesine katkı sağlamaktadır.

Bölümü: Veri Bilimi ve Analitiği

Hazırlayan: Fatih OKUMUŞ

Öğrenci Numarası: 20253005014

Tarih: 06.12.2025

Doktor Öğretim Üyesi: Ertan AKGENÇ