



**BTK**  
**AKADEMİ**

# Programlama

Doç. Dr. Zafer CÖMERT



# Bölüm 4

## Akış Diyagramları

# Giriş

## İçerik

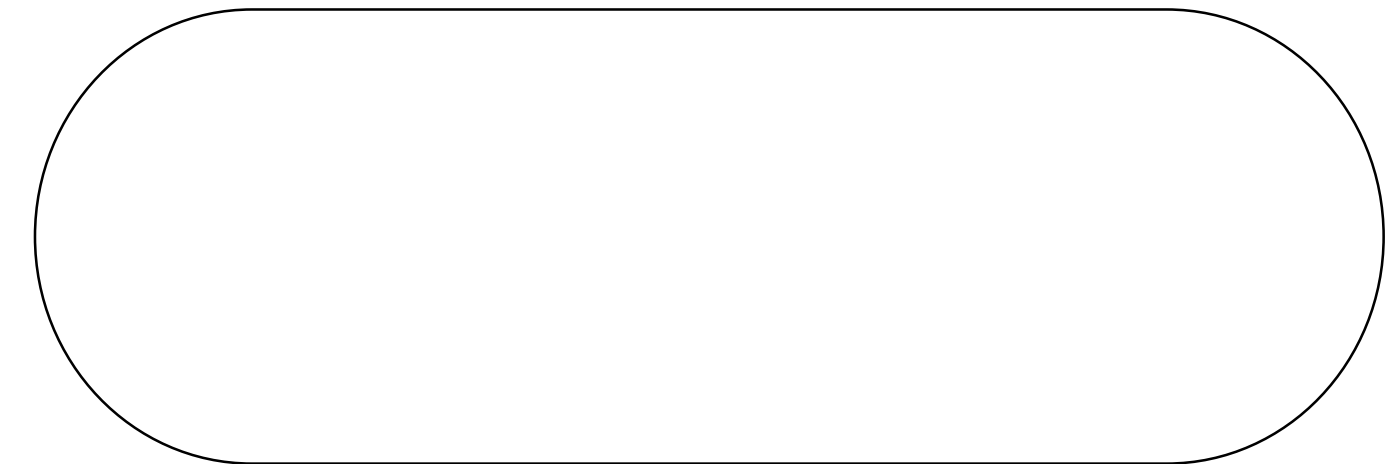
- Akış Diyagramı Nedir?
- Akış Diyagramı Sembolleri
- Akış Diyagramı Oluşturma Kuralları
- Basit Akış Diyagramı Uygulamaları
- Örnekler ve Alıştırmalar

# Akış Diyagramı Nedir?

- Akış diyagramı, bir problemin çözümü için tanımlanan algoritma adımlarını ve bu adımlar arasındaki ilişkileri görsel olarak gösteren bir şemadır.
- Başka bir deyişle, yapılacak işlemleri belirli sembollerle adım adım gösteren ve her adımı birbirine oklarla bağlayan bir diyagramdır.
- Bu sayede karmaşık işlemler bile daha anlaşılır ve düzenli bir biçimde ifade edilebilir.

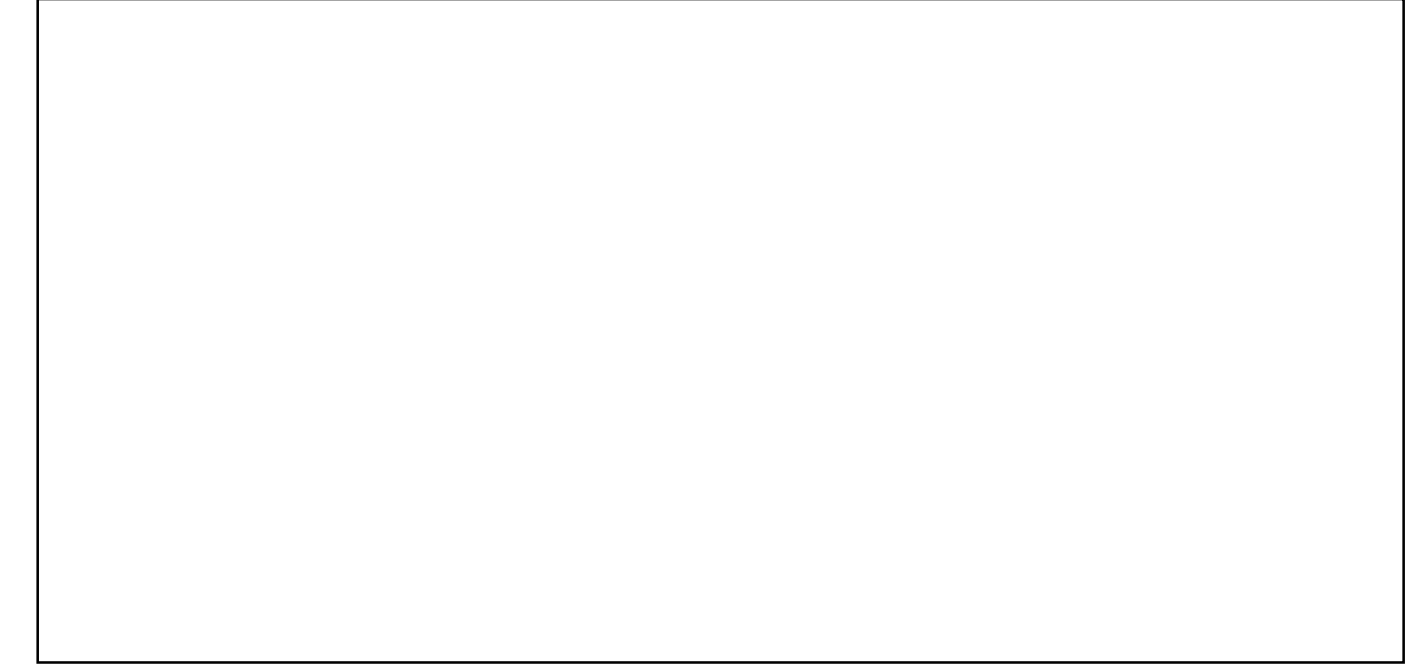
# Terminal Start/End

- Terminal sembolü, bir akış diyagramında başlangıç ve bitiş noktalarını göstermek için kullanılır.
- Genellikle oval (elips) şekliyle temsil edilir ve içinde "START" veya "STOP" gibi ifadeler yer alır.



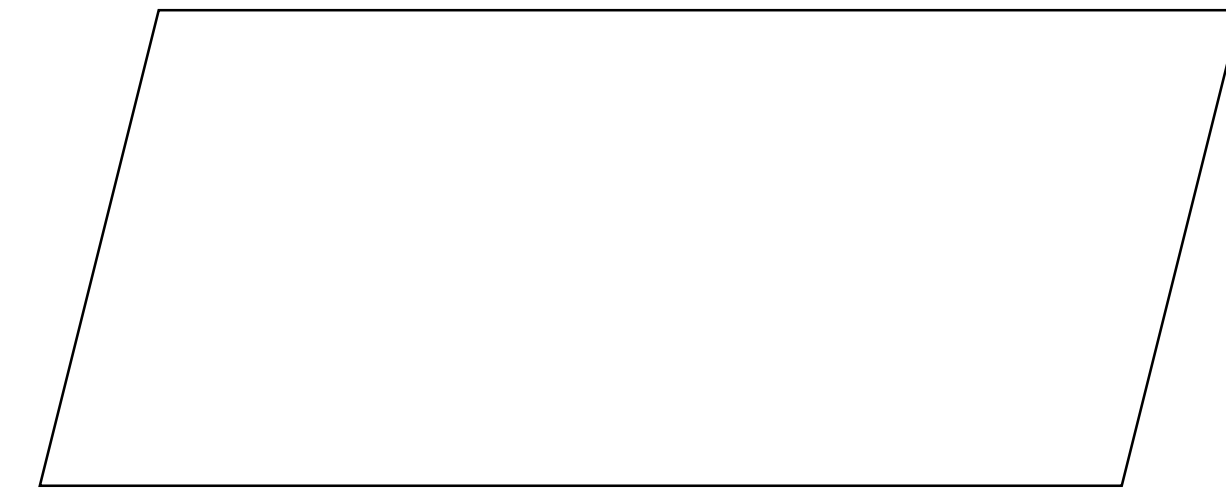
# İşlem Processing

- İşlem sembolü, akış diyagramlarında bir işlem adımını temsil eder ve genellikle dikdörtgen şeklindedir.
- Bu sembol, bilgisayarın veri üzerinde yaptığı hesaplama, atama veya karar dışı tüm işlemleri gösterir.



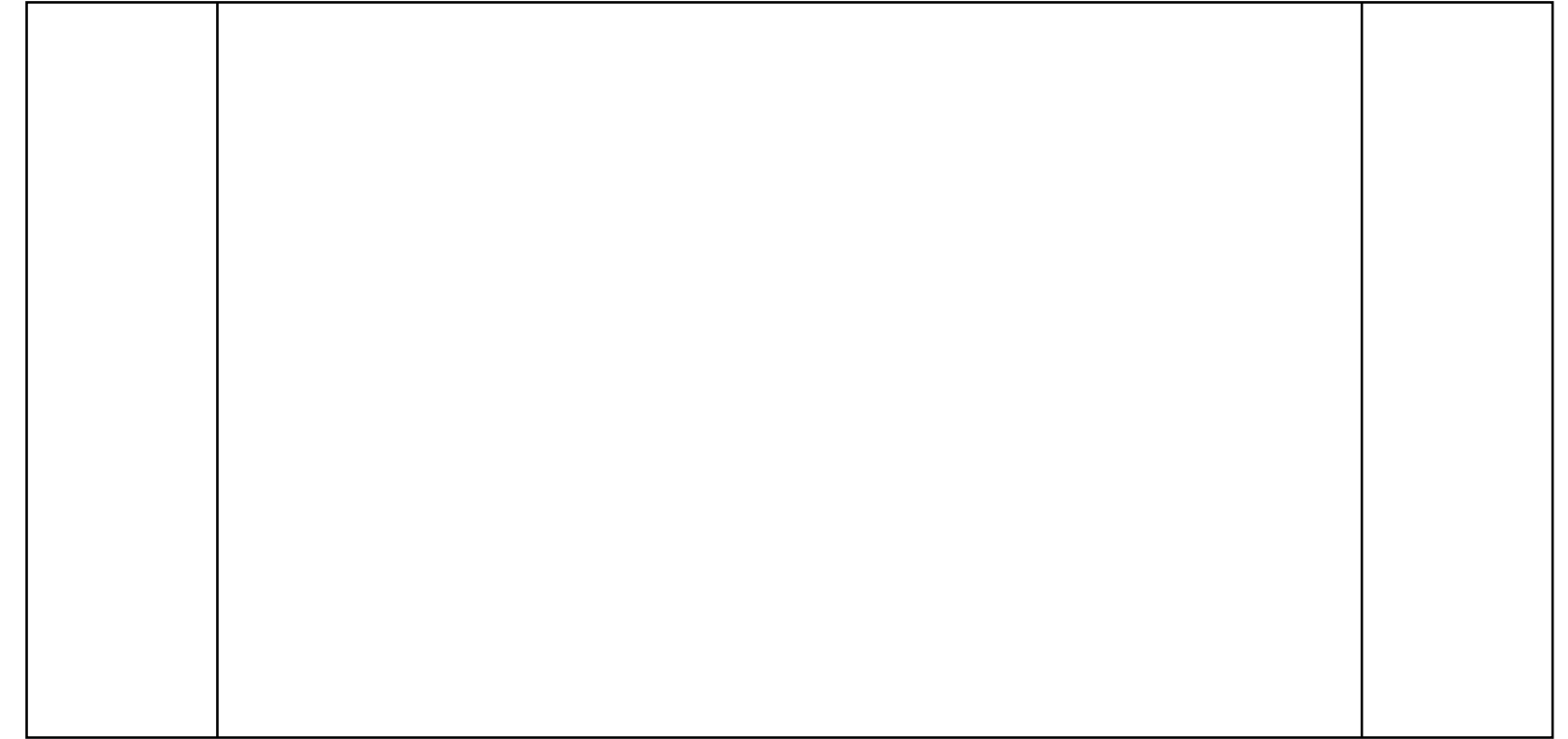
# Giriş / Çıkış Input/Output

- Giriş/Çıkış sembolü, bir programda veri alma (giriş) veya sonuç gösterme (çıkış) işlemlerini temsil eder.
- Paralelkenar şeklinde çizilir ve içinde genellikle “INPUT” ya da “PRINT” gibi ifadeler bulunur.



# Önceden Tanımlı İşlem Predefined Processing

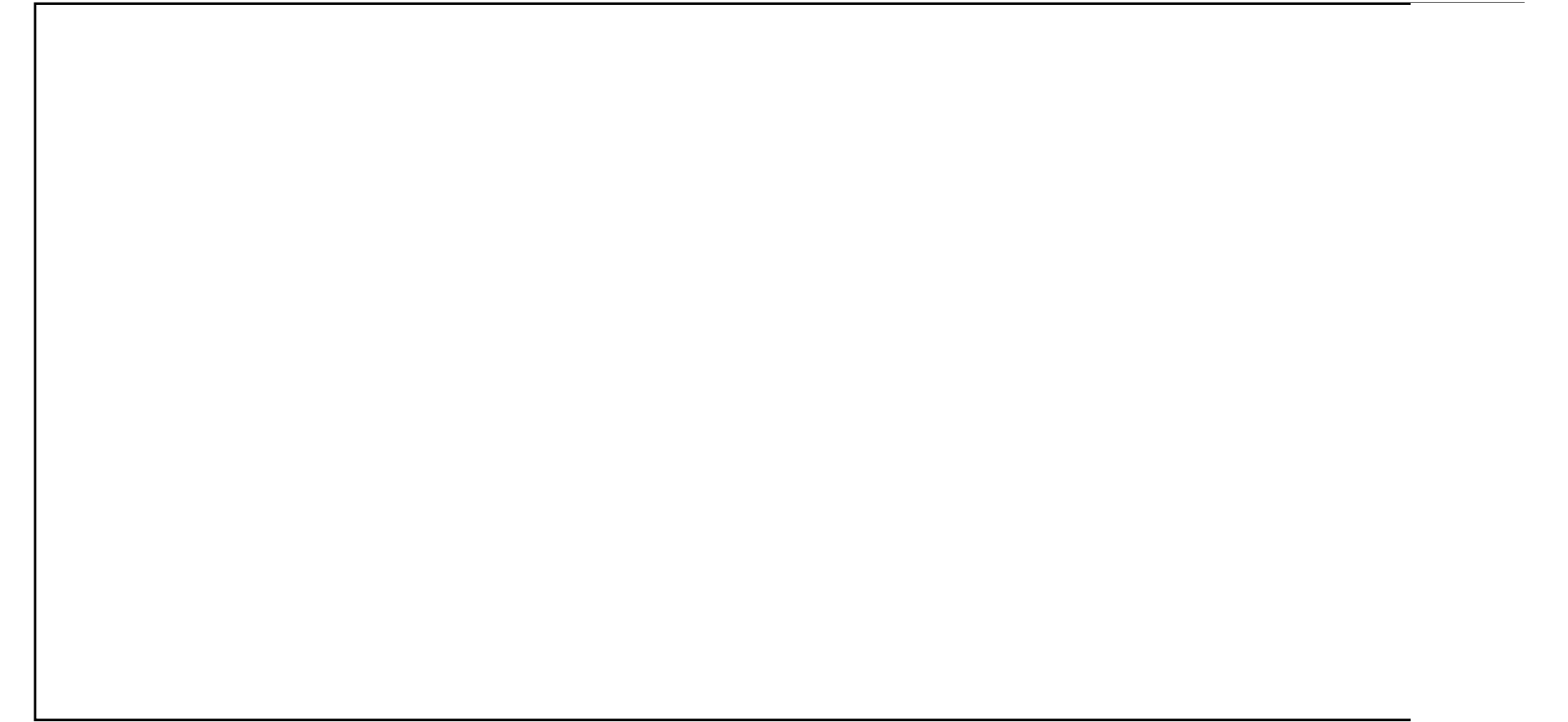
- Akış diyagramlarında alt program ya da fonksiyon çağrısını temsil eder ve çift çizgili dikdörtgen ile gösterilir.
- Bu sembol, detayları diyagramda gösterilmeyen ancak başka bir yerde tanımlı olan işlemlerin özet olarak gösterilmesini sağlar.





# Yorum Comment

- Comment (Açıklama) sembolü, akış diyagramlarında bir adım hakkında ek bilgi veya not vermek için kullanılır.
- Genellikle köşeleri açık dikdörtgen veya parantezli kutu şeklinde çizilir ve algoritmanın anlaşılmasını kolaylaştırmak amacıyla açıklayıcı metin içerir.



# Akış Çizgisi Flow line

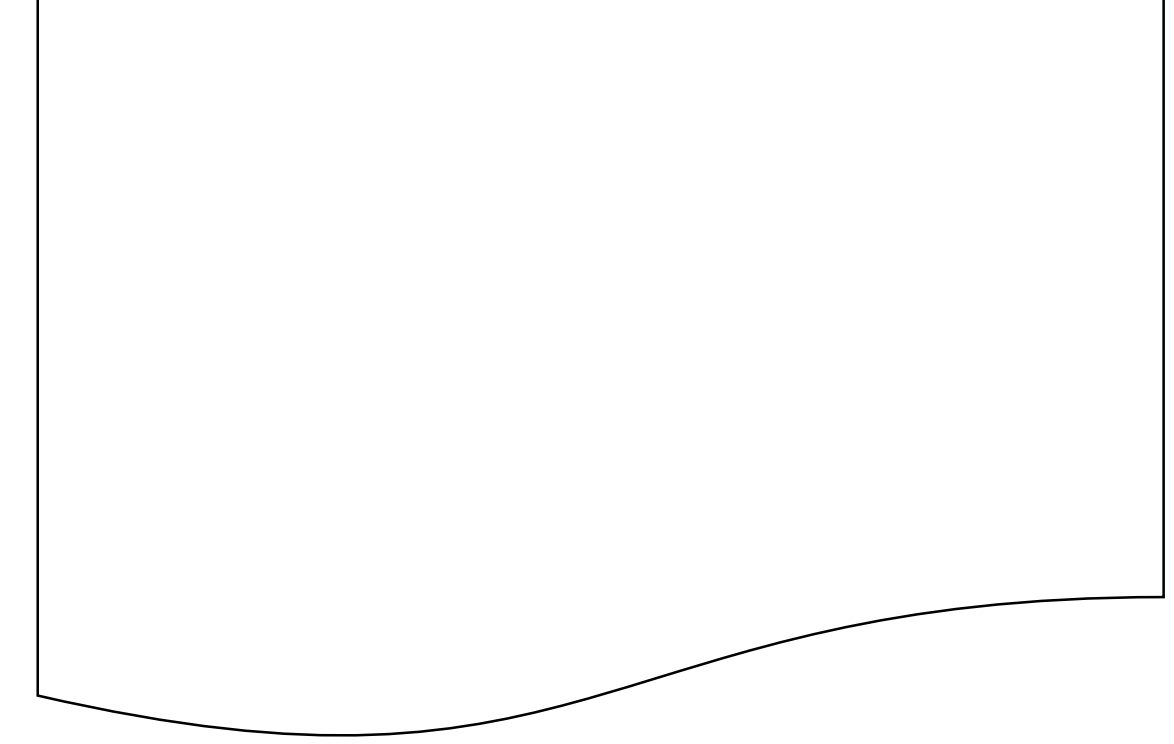
- Flow line (Akış çizgisi), akış diyagramında işlem sırasını ve yönünü gösteren ok şeklindeki çizgidir.
- Bu çizgiler, bir sembolden diğerine geçişi belirleyerek algoritmanın adım adım nasıl ilerlediğini görsel olarak ifade eder



# Belge Giriş/Çıkış

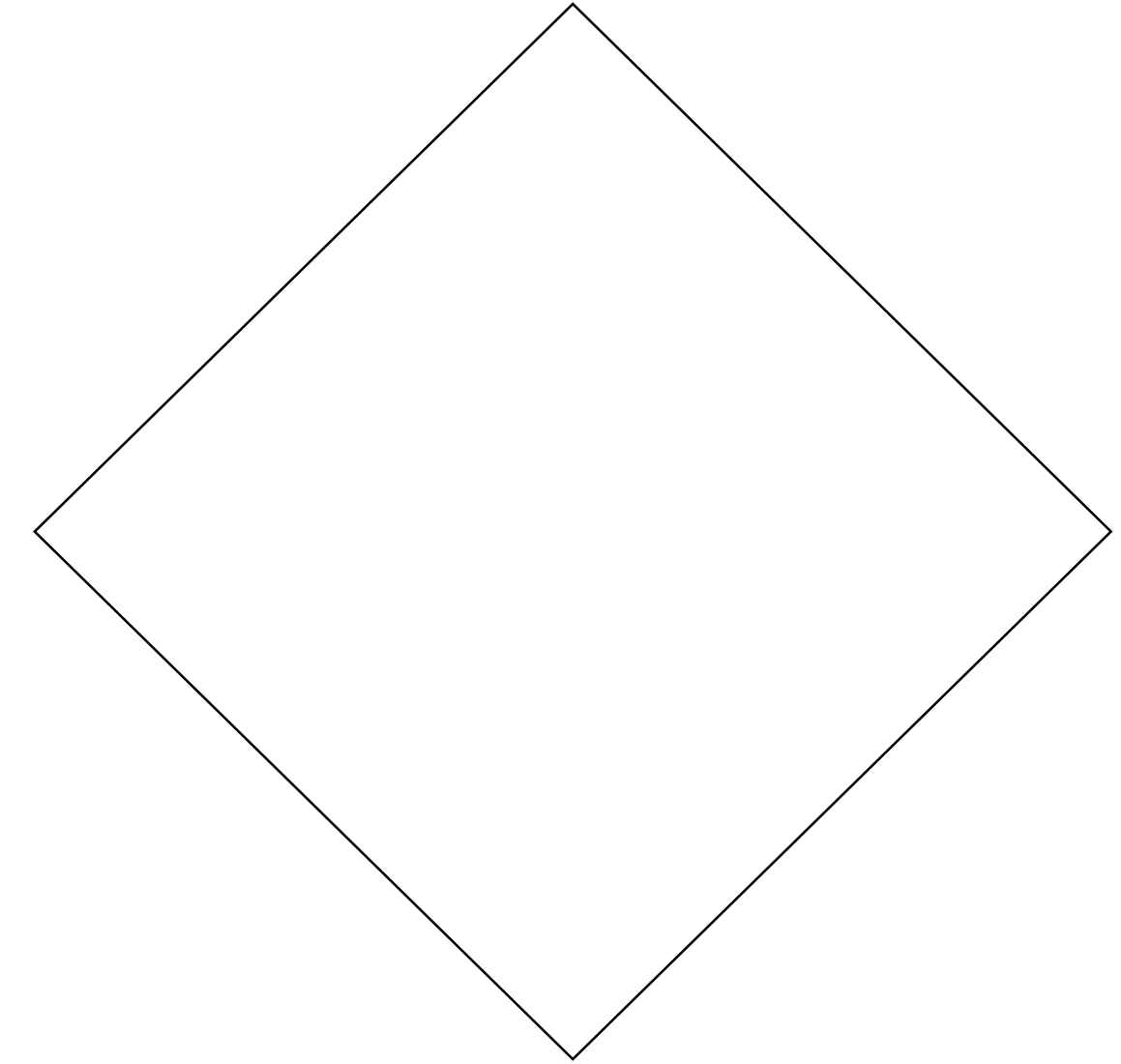
## Document Input/Output

- Belge Giriş/Çıkış sembolü, verinin bir belgeden alınması veya bir belgeye yazılması durumlarını gösterir.
- Genellikle üst kenarı dalgalı bir dikdörtgen olarak çizilir ve çıktı raporları veya fiziksel belgelerle etkileşimi simgeler.



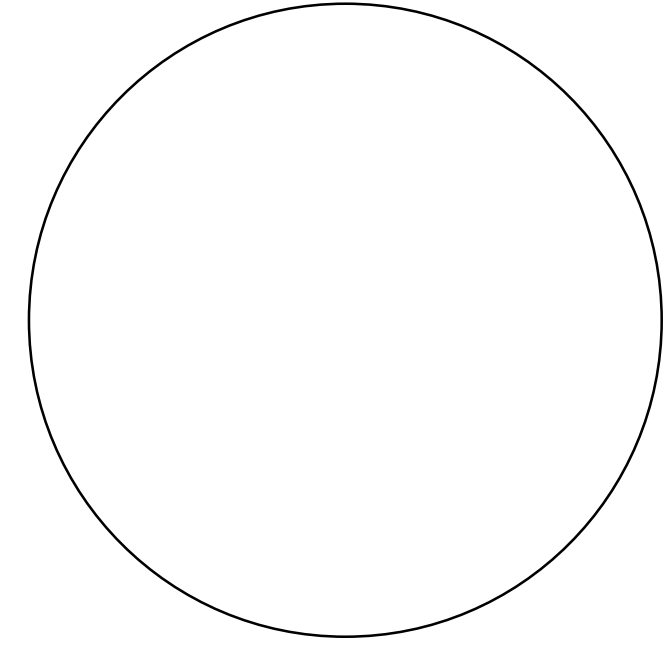
# Karar Decision

- Karar sembolü, bir koşulun doğru veya yanlış olup olmadığını kontrol etmek için kullanılır.
- Eşkenar dörtgen (baklava) şeklindedir ve genellikle "Evet/Hayır" (Yes/No) dallanmaları ile sürecin yönünü belirler.



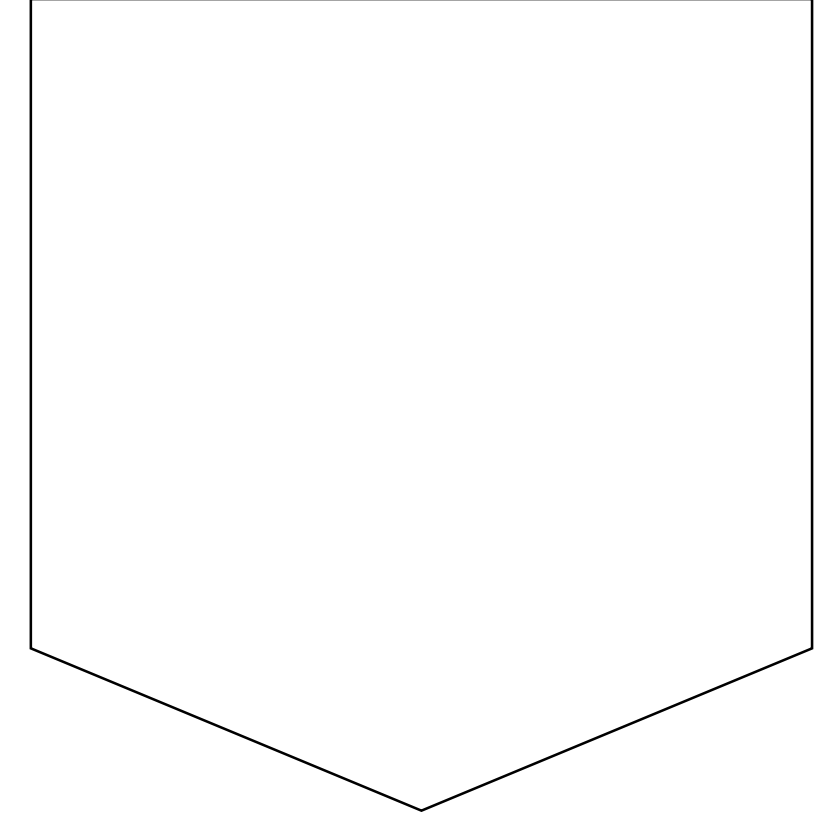
# Sayfa İçi Bağlayıcı On-Page Connector

- Sayfa İçi Bağlayıcı sembolü, aynı sayfa üzerindeki farklı akış diyagramı bölümlerini birbirine bağlamak için kullanılır.
- Küçük daire şeklindedir ve karmaşık diyagramlarda okunabilirliği artırmak amacıyla kullanılır.



# Sayfa Dışı Bağlayıcı Off-Page Connector

- Sayfa Dışı Bağlayıcı sembolü, akış diyagramının başka bir sayfada devam ettiğini göstermek için kullanılır.
- Genellikle yatay bir beyzbol sahası (ev plakası) şeklindedir ve çok sayfalı diyagramlarda akışın izlenmesini kolaylaştırır.



# Akış Diyagramı Oluşturma Kuralları

Tek giriş ve çıkış noktası

Akış yönü ve sırası

Çizgilerin kesişmemesi

Sembollerin düzeni ve boşlukları

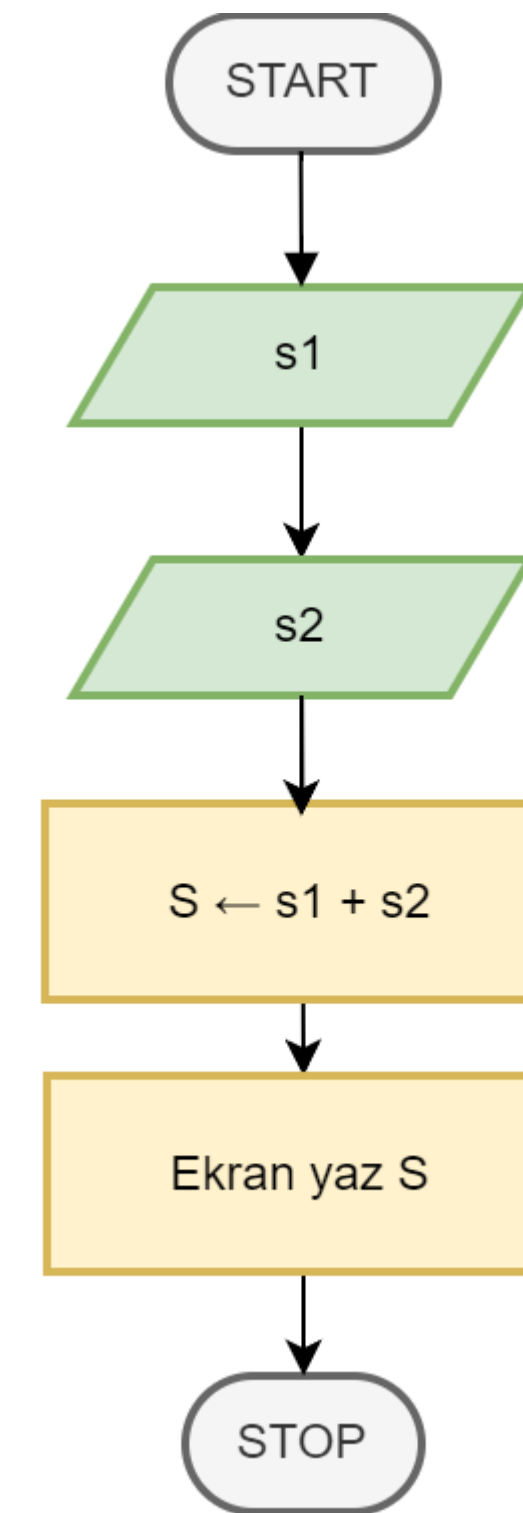
Doğru sembol kullanımı

Karar noktalarında dallanma

Okunabilirlik ve yorumlar

# İki Sayının Toplamı

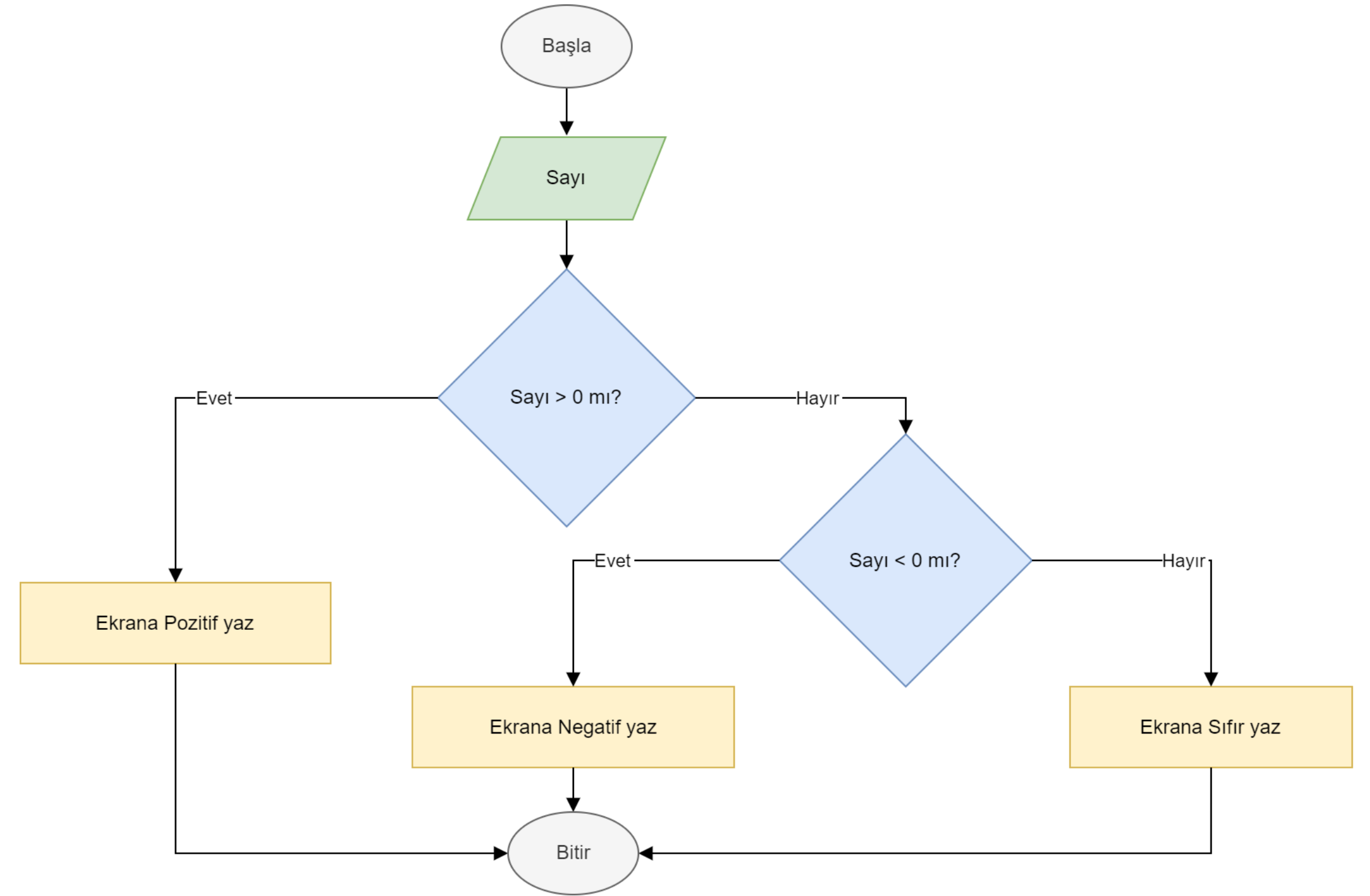
1. Başla.
2. İlk sayıyı kullanıcıdan al.
3. İkinci sayıyı kullanıcıdan al.
4. İki sayıyı topla ve sonucu bir değişkende sakla.
5. Toplam sonucunu ekranda göster.
6. Bitir.





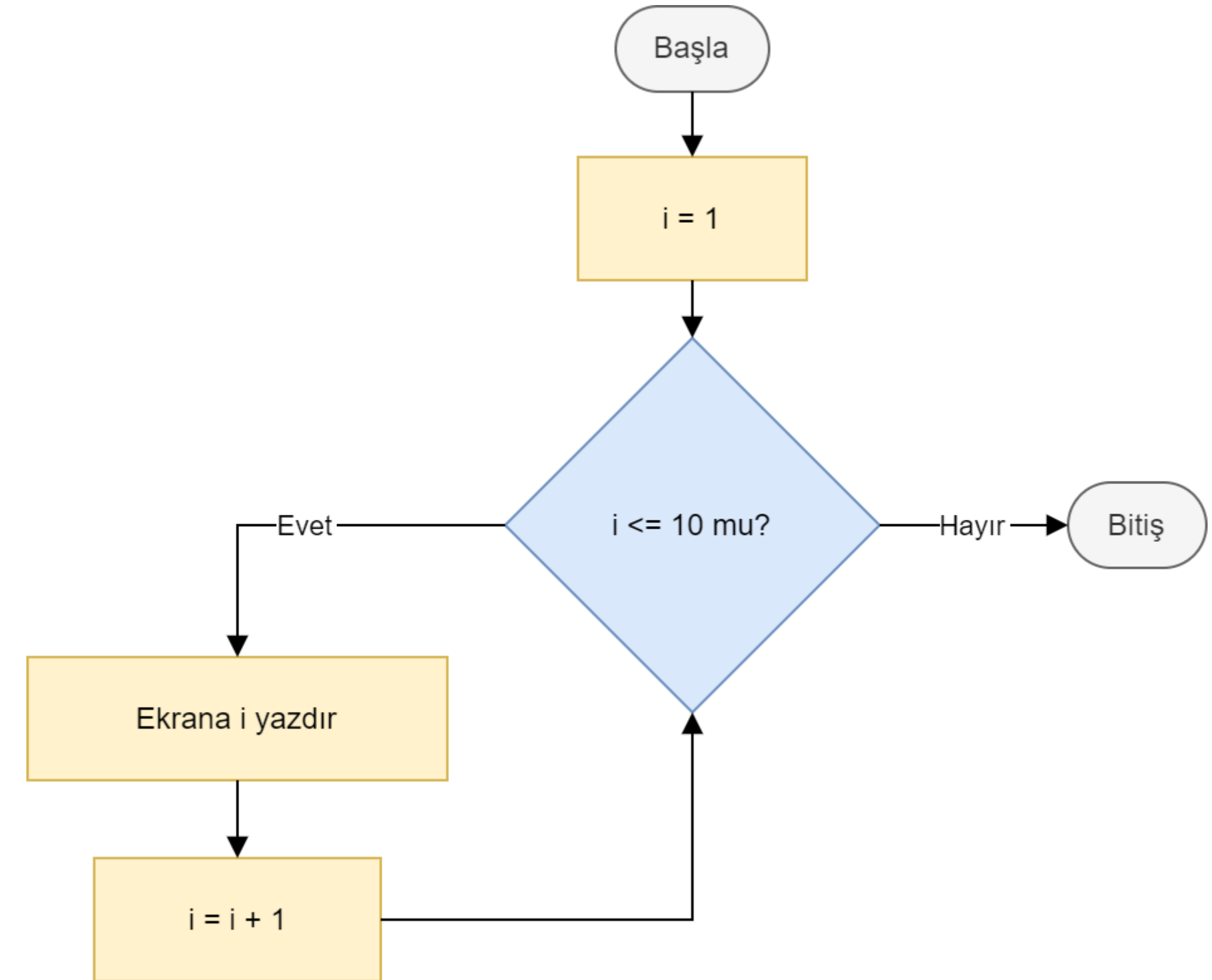
# Pozitif/Negatif Kontrolü

1. Başla.
2. Kullanıcıdan bir sayı al.
3. Eğer sayı  $> 0$  ise pozitif olduğunu bildir.
4. Eğer sayı  $< 0$  ise negatif olduğunu bildir.
5. Eğer sayı 0 ise sayının sıfır olduğunu (pozitif/negatif olmadığını) bildir.
6. Bitir.

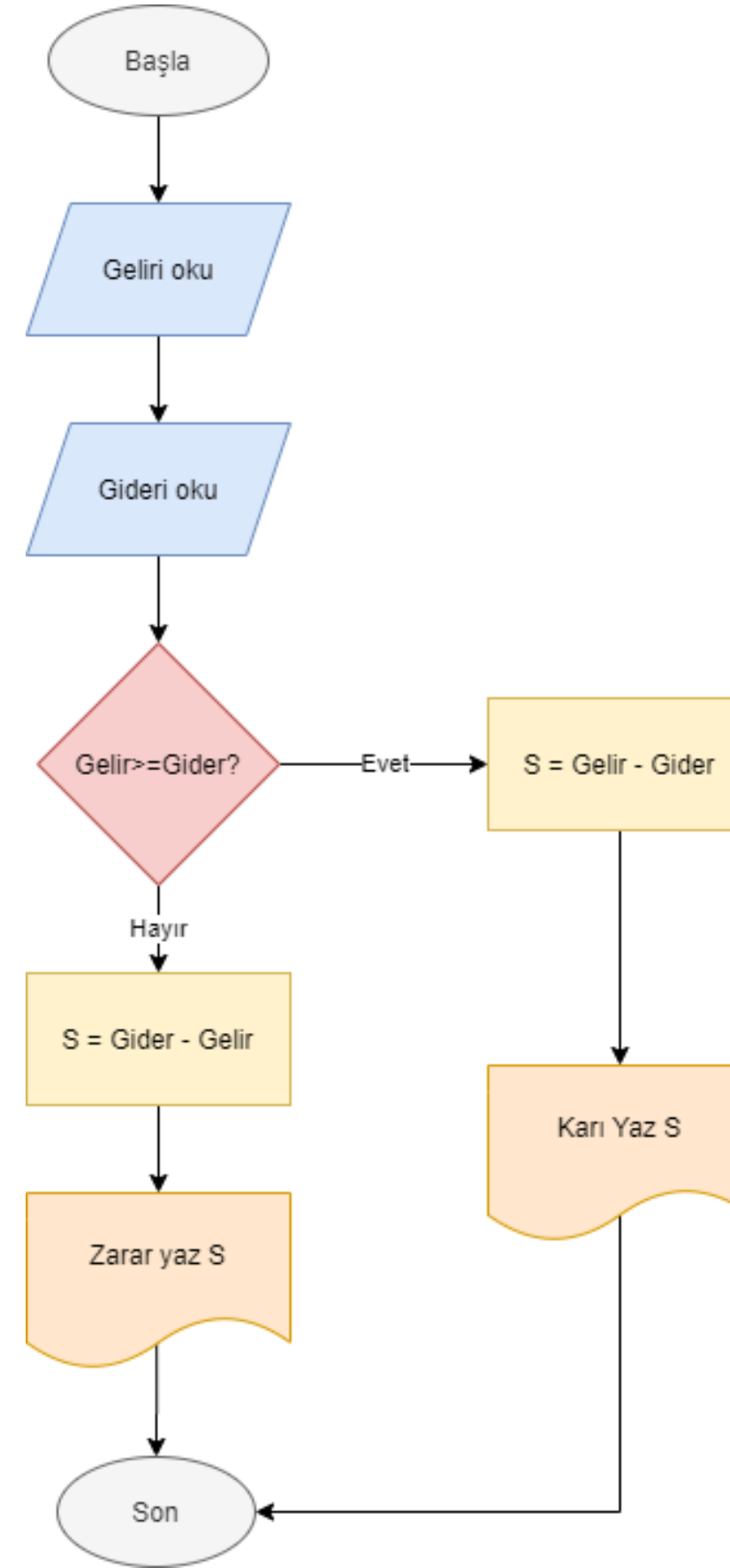


# 1'den 10'a Kadar Sayıları Yazdırma

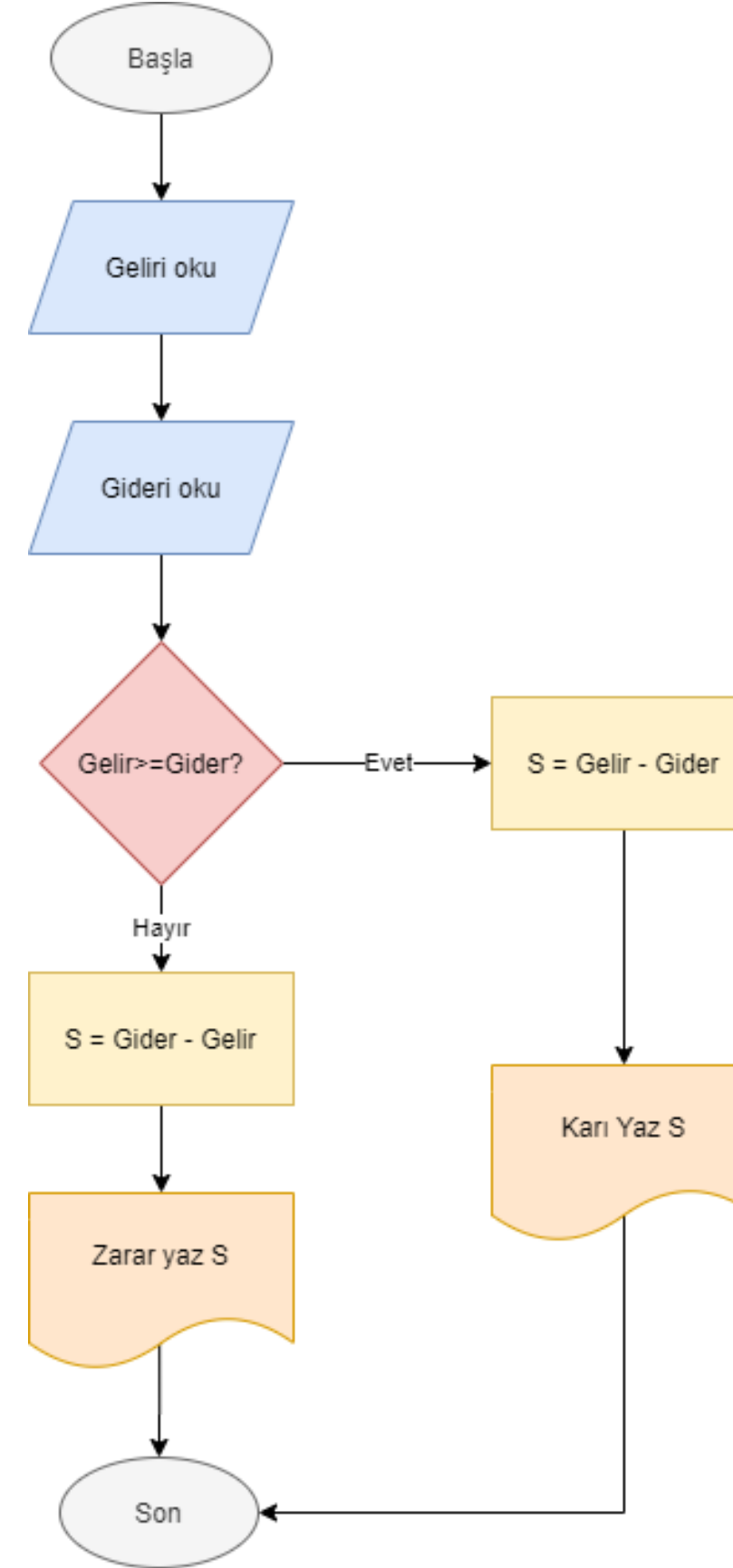
1. Başla.
2. Bir sayaç değişkeni  $i$  değerini 1 olarak ata.
3.  $i$  değeri 10'dan küçük veya eşit olduğu sürece aşağıdakileri tekrarla:
  1.  $i$  değerini ekrana yazdır.
  2.  $i$  değerini 1 artır.
4.  $i$  değeri 10'dan büyük olduğunda döngü sona erer.
5. Bitir.



# Gelir - Gider

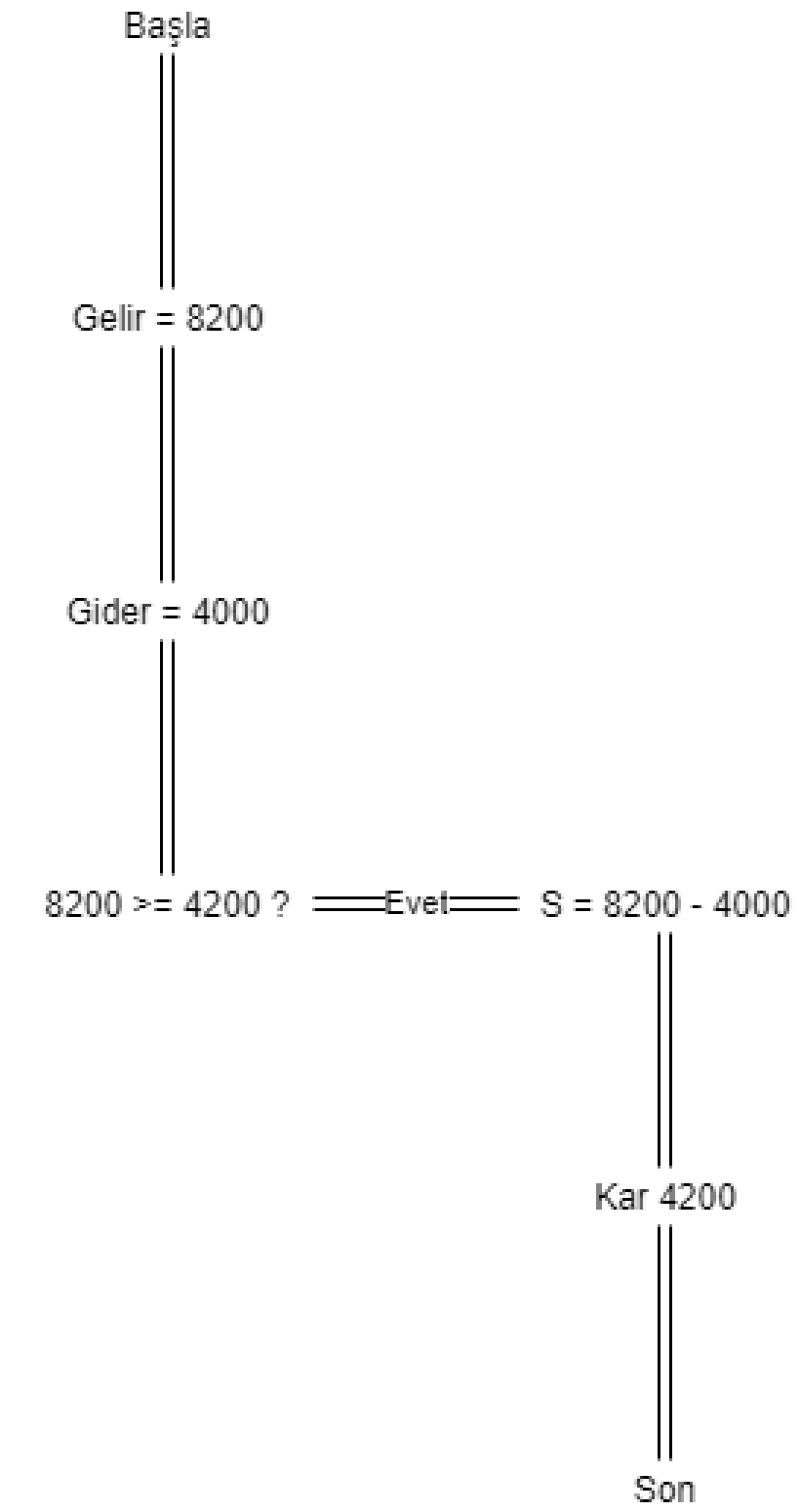
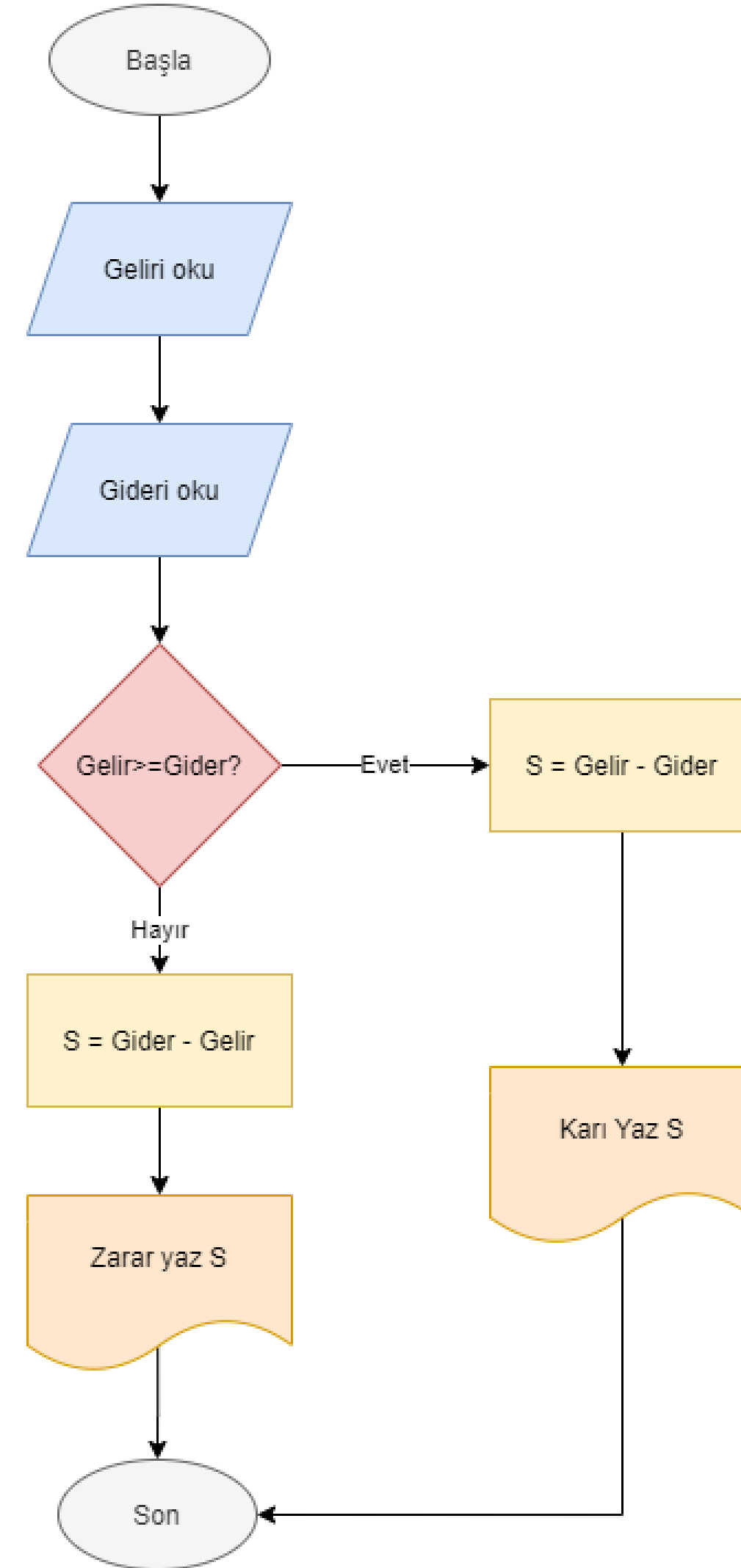


# Gelir - Gider

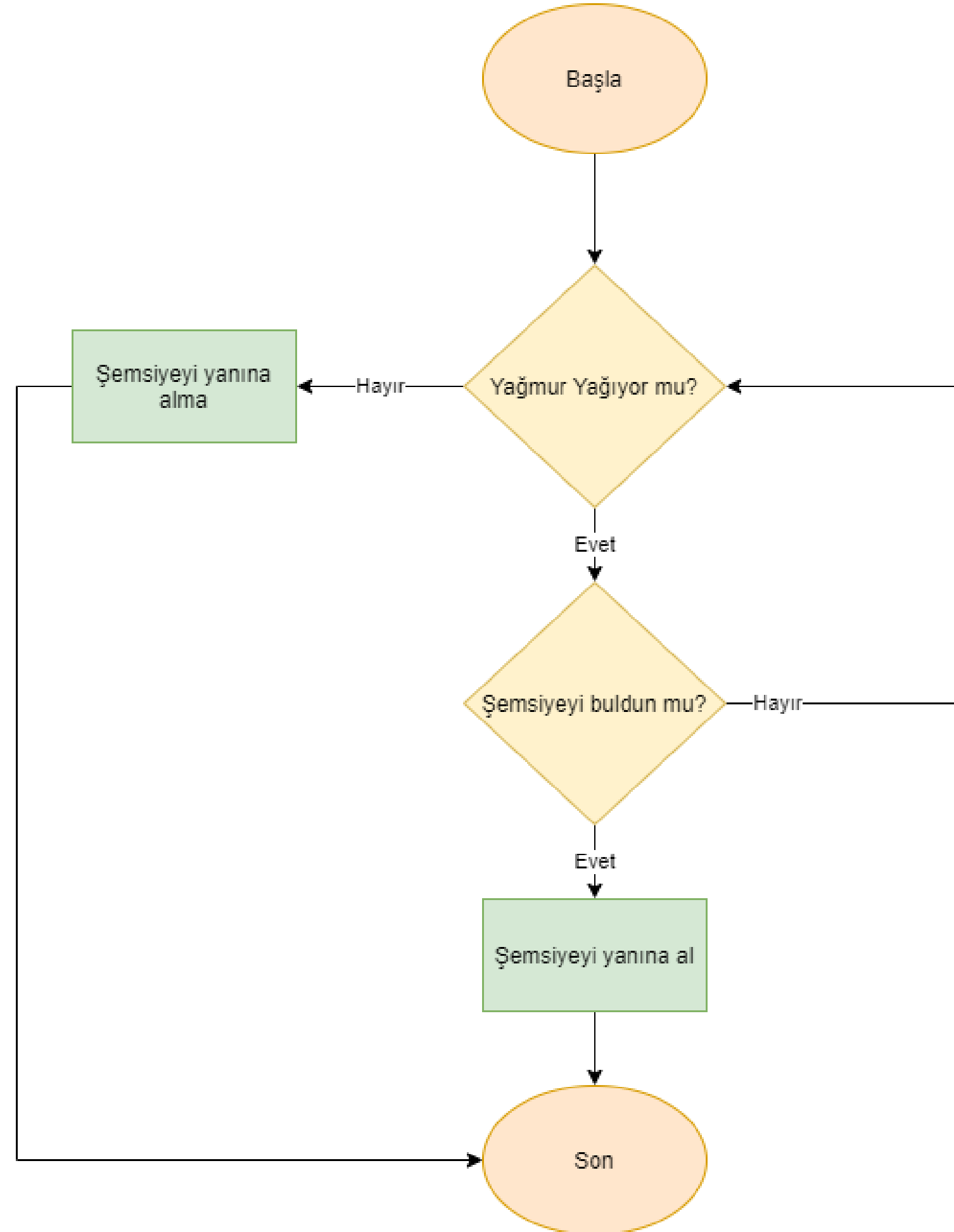


Başla  
||  
Gelir = 5000  
||  
Gider = 7500  
||  
5000 >= 7500?  
||  
S = 7500 - 5000  
||  
Zarar = 2500  
||  
Son

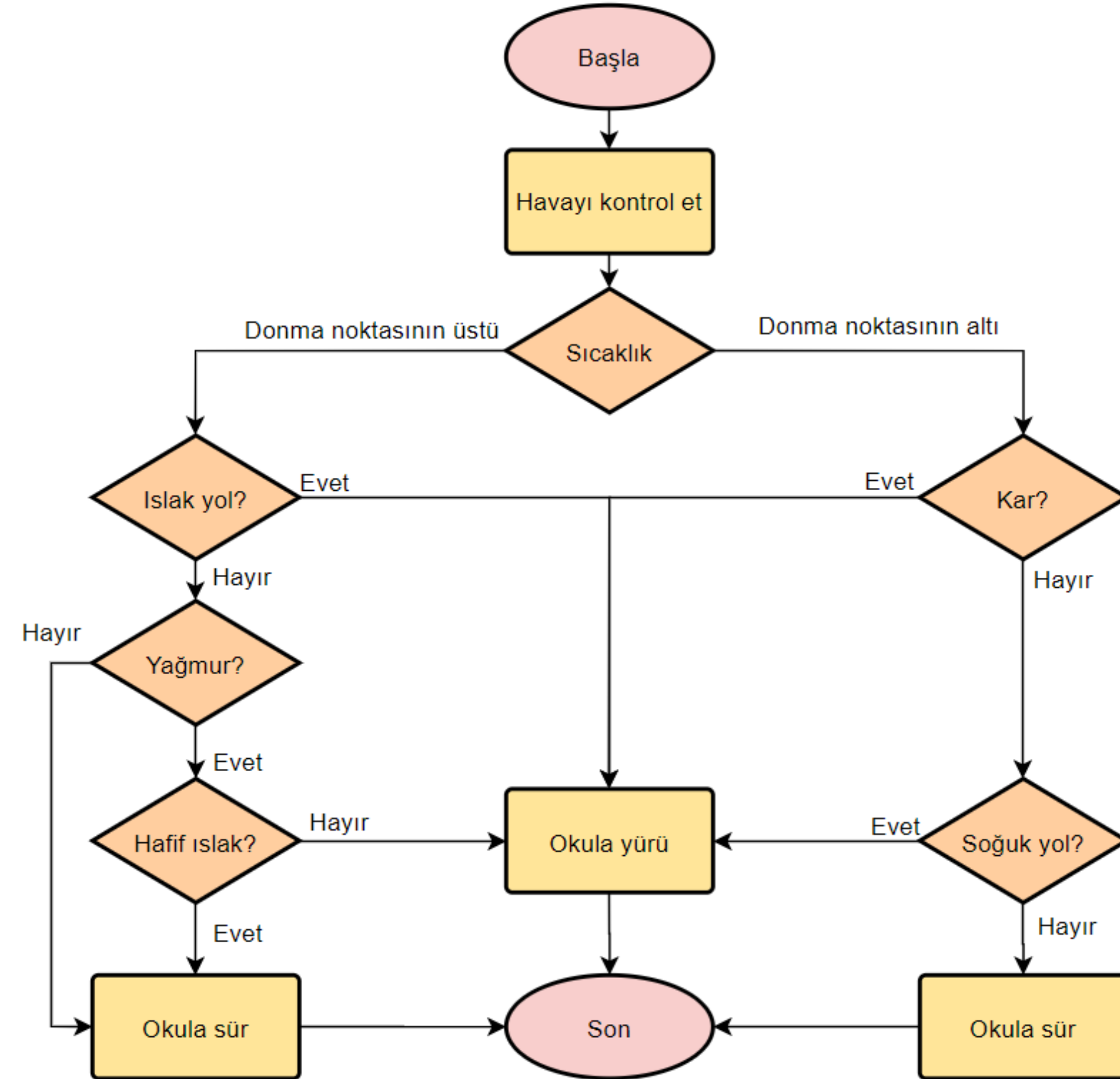
# Gelir - Gider



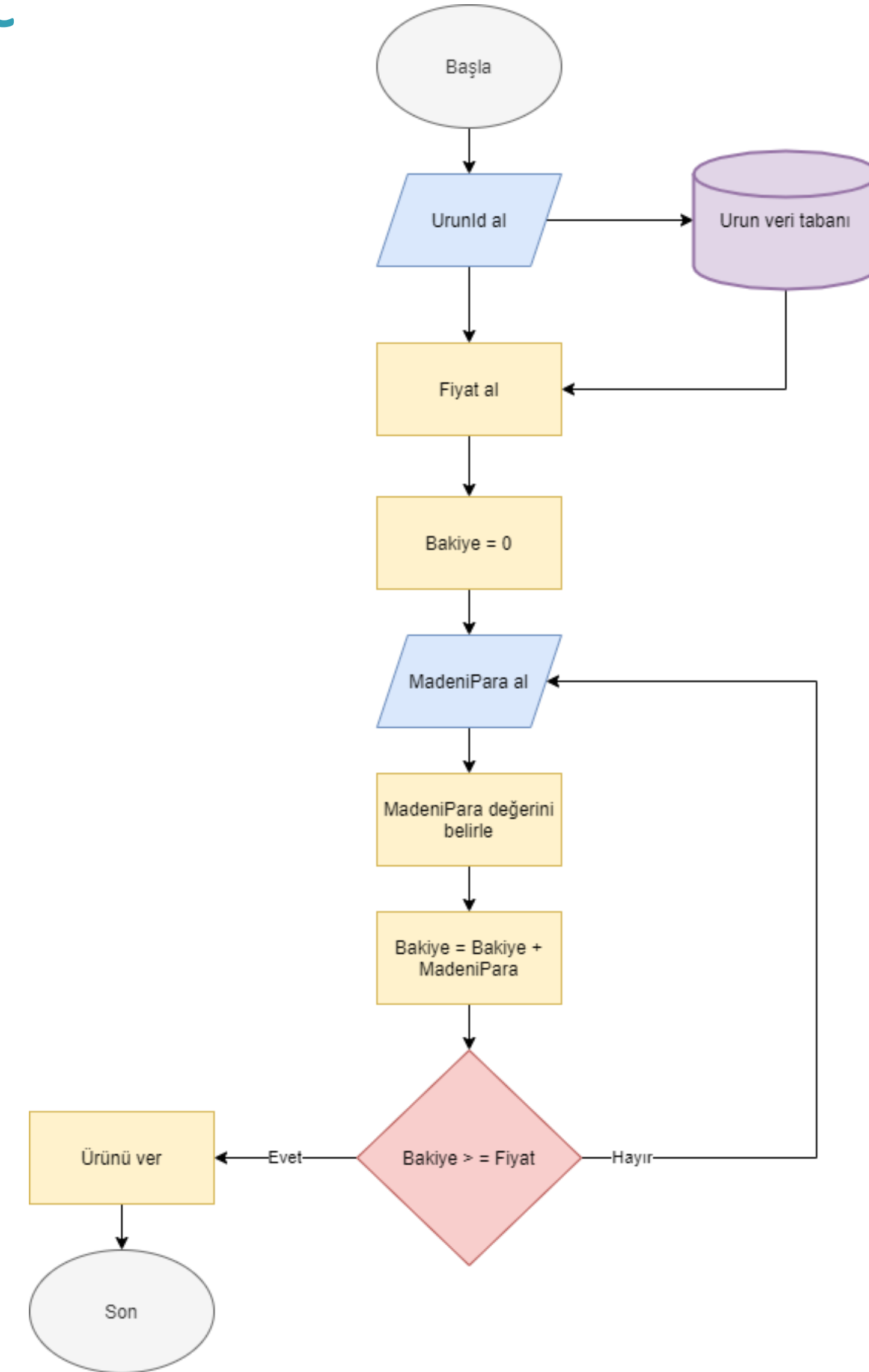
# Şemsiye Alma



# Bisiklet?

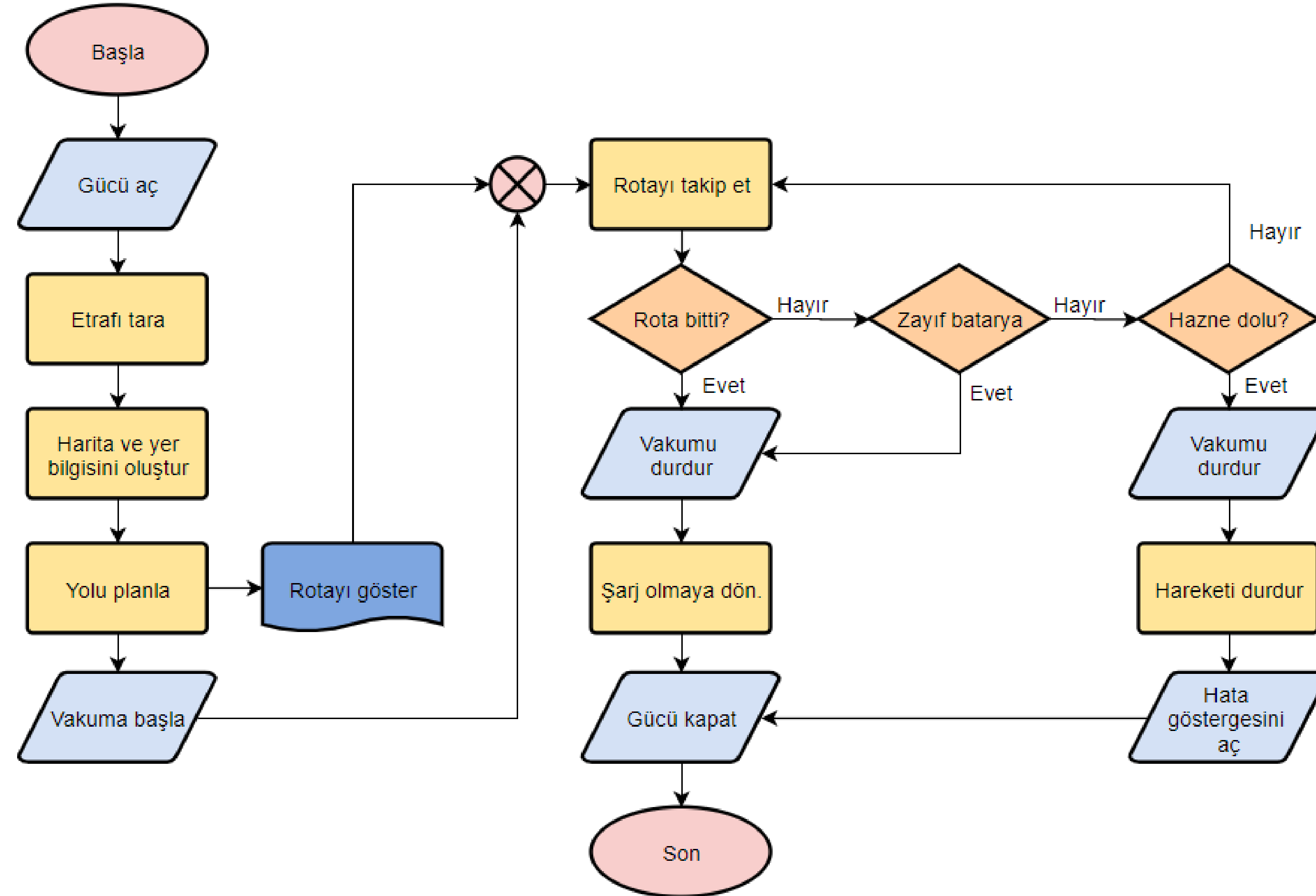


# Otomat





# Robot Süpürge



# Teşekkürler

ZAFER CÖMERT  
Öğretim Üyesi