



**BTK**  
**AKADEMİ**

# Programlama

Doç. Dr. Zafer CÖMERT



# Bölüm 7

## Döngü Yapıları

# Giriş

## İçerik

- Giriş
- while döngüsü
- for döngüsü
- range() Fonksiyonu
- break deyimi
- continue deyimi
- İç içe döngüler

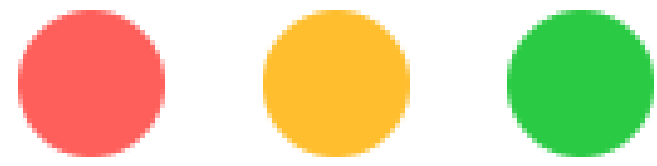
# Döngüler

- Programlamada döngüler, belirli bir kod bloğunu birden çok kez çalıştırmayı sağlayan yapılar olarak kritik bir öneme sahiptir.
- Birçok programlama probleminde, bazı işlemlerin tekrar tekrar yürütülmesi gerekir.

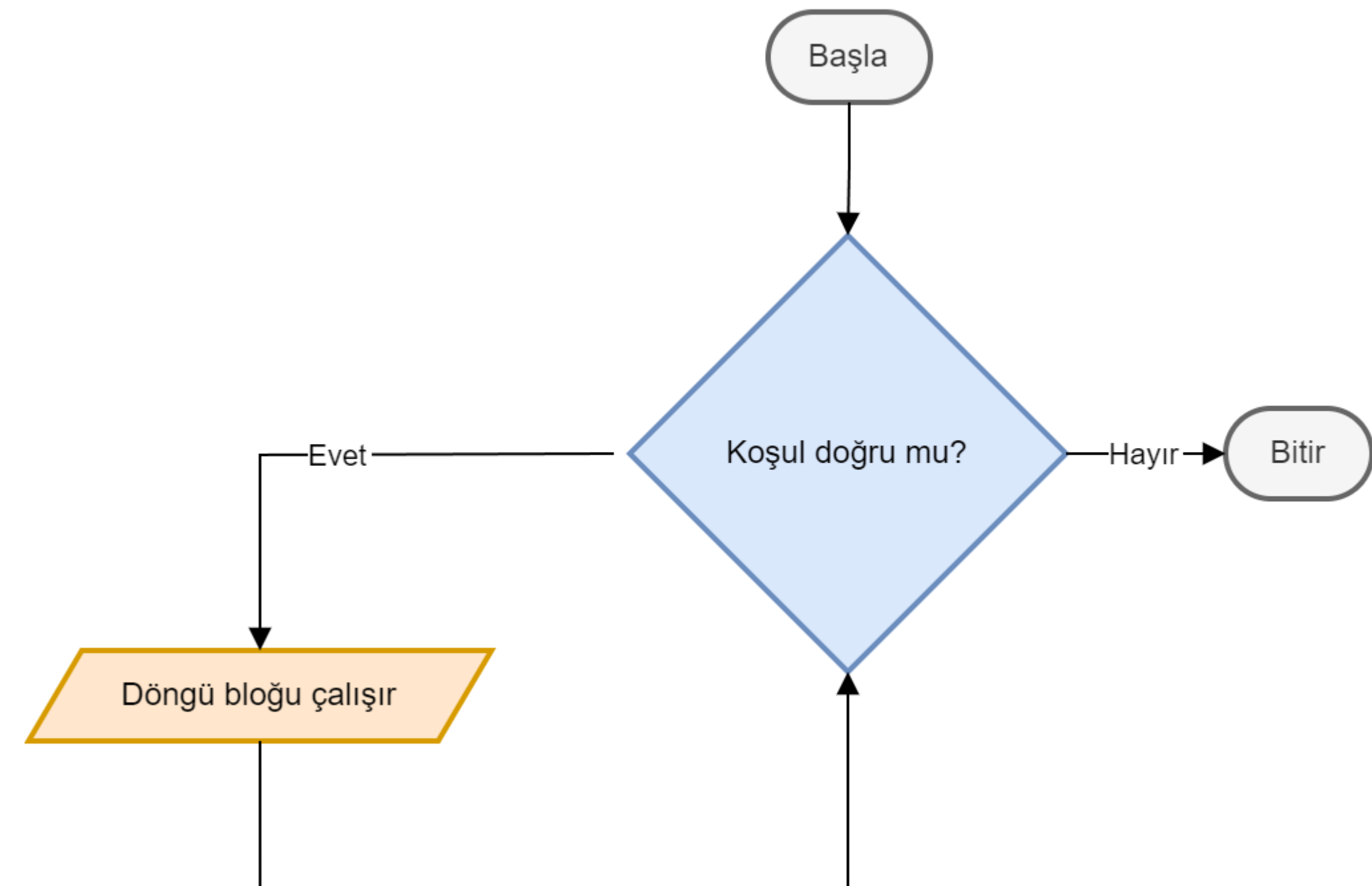
Koşul Kontrollü  
Döngüler

Sayaç Kontrollü  
Döngüler

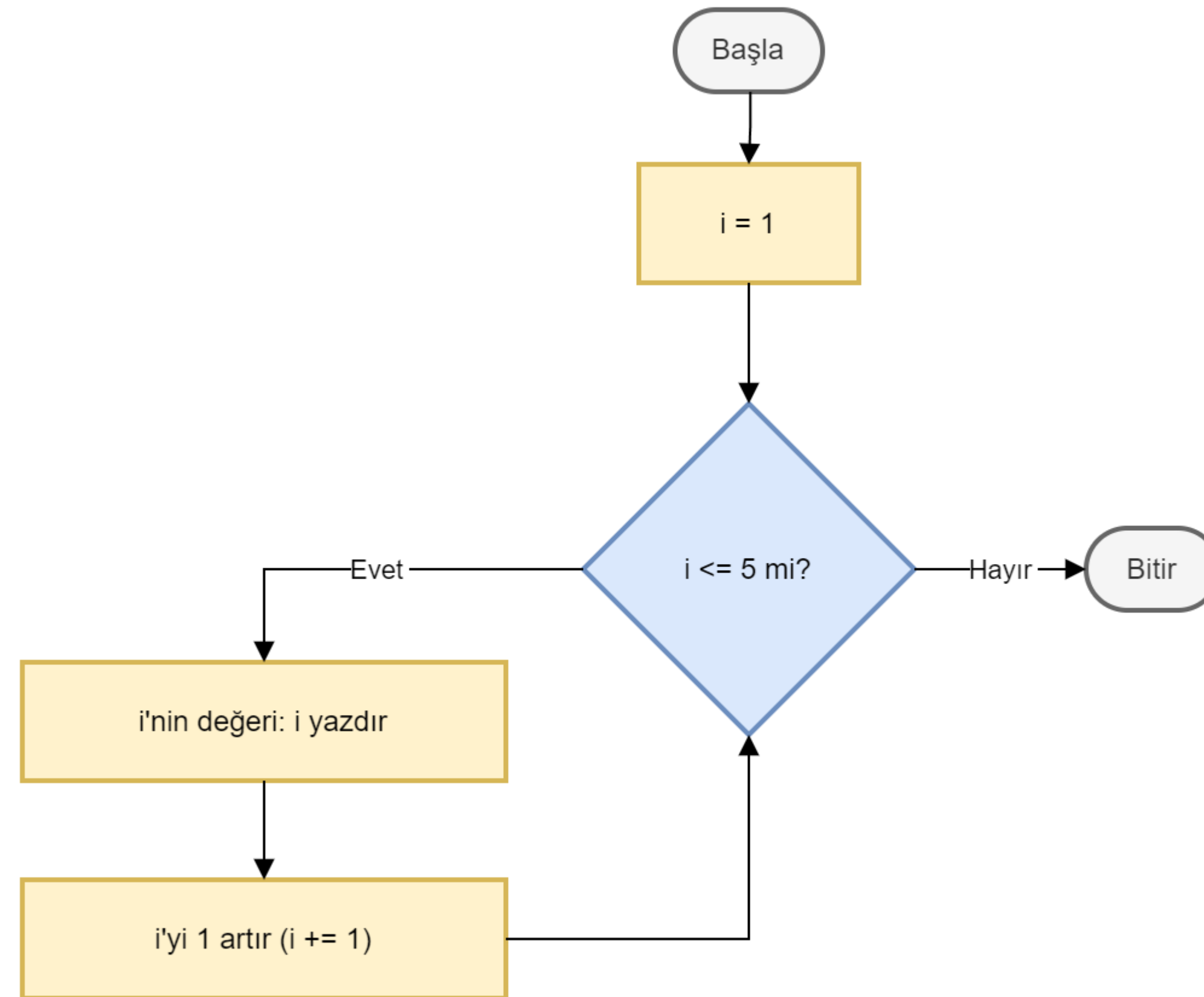
# while



```
1  kosul = False
2  while kosul:
3      # Döngü bloğu
4      pass
```



# while



# for



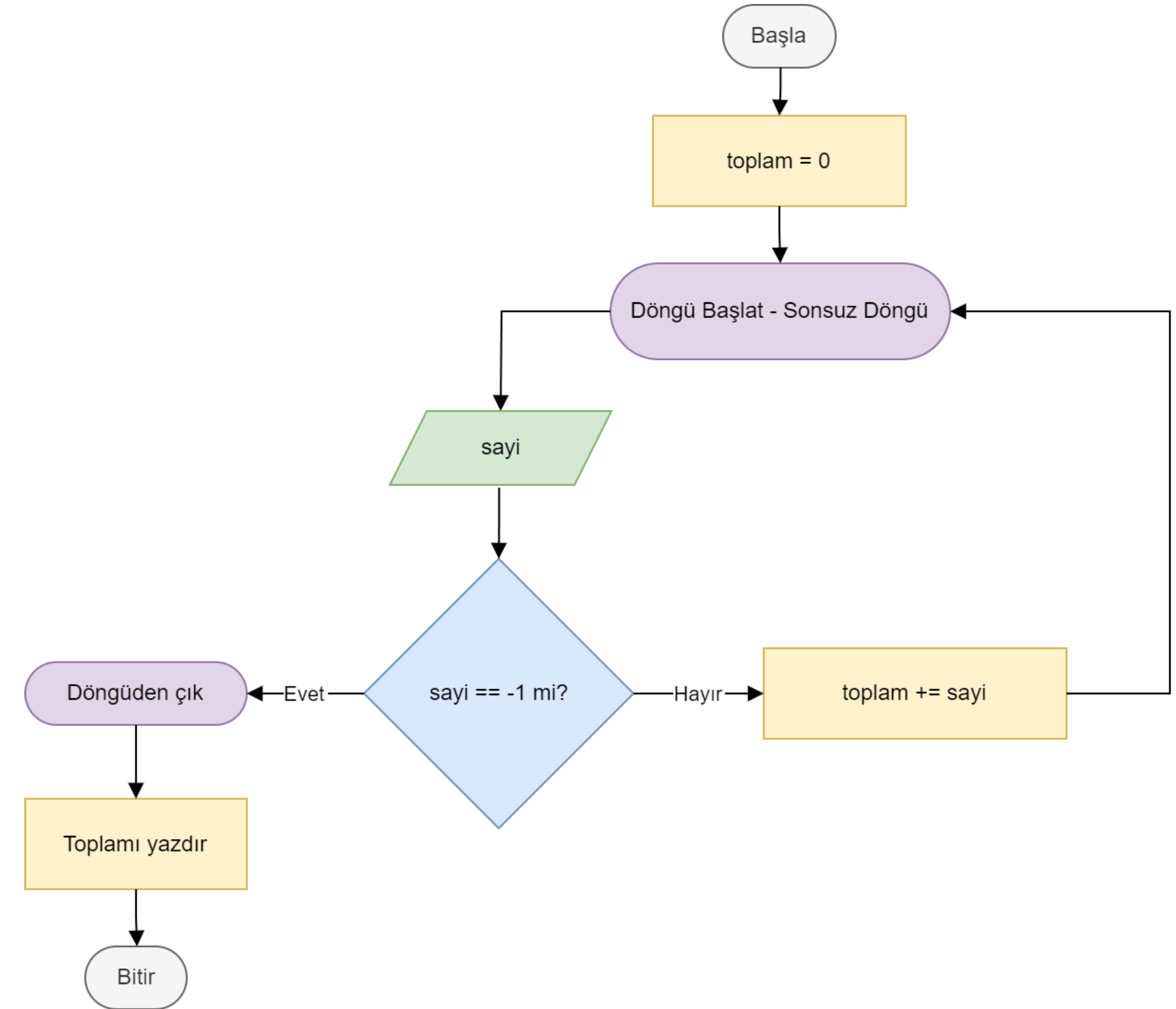
```
1  for dongu_degiskeni in yineleme_yapilacak_veri:
2      # Döngü bloğu
3      pass
```



```
1  for i in range(1, 6):
2      print(i)
```

# break

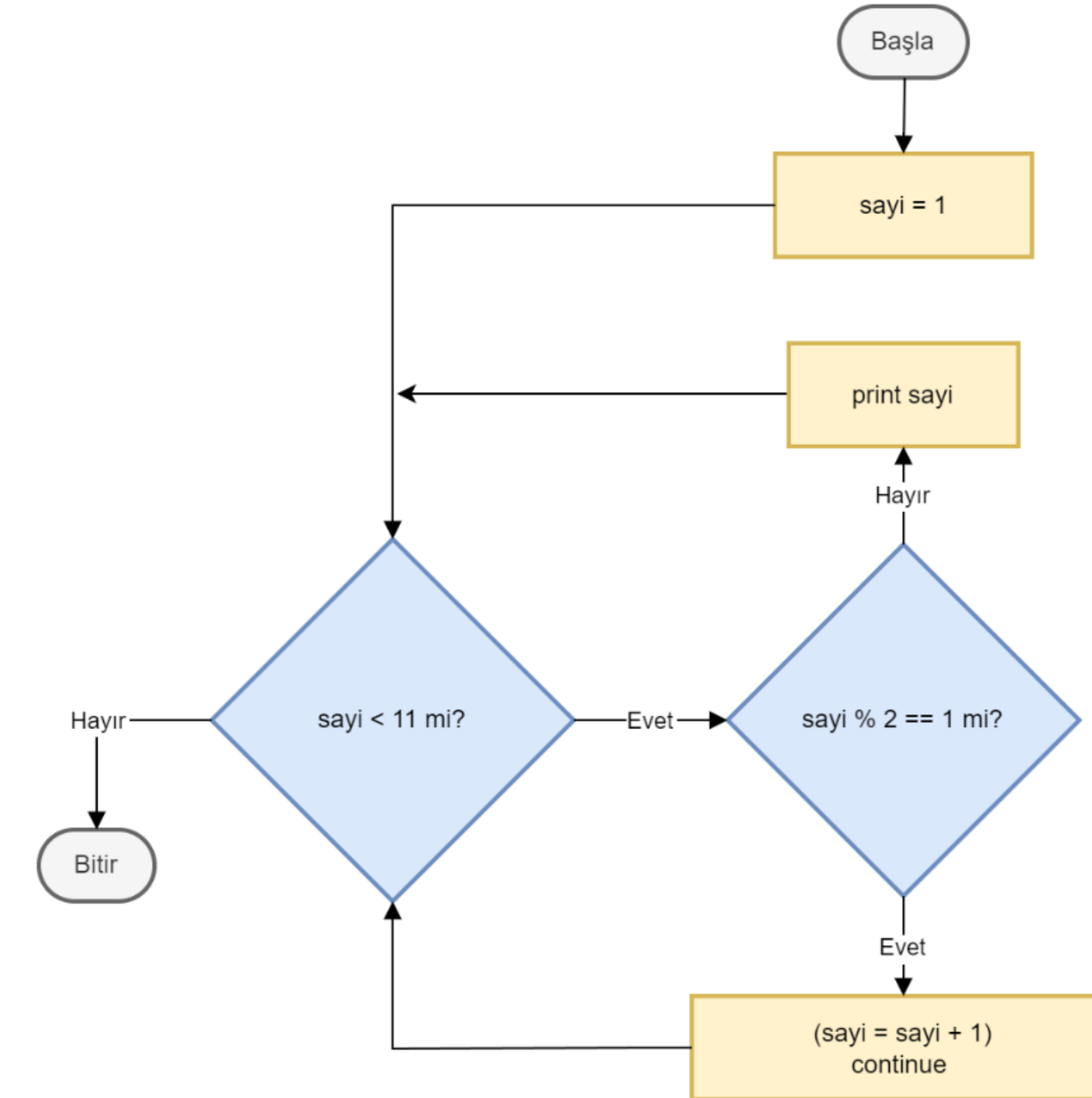
- İçinde bulunduğu döngüyü anında sonlandırmayı sağlar.
- Program akışı, break ile karşılaşılan anda döngü bloğundan çıkar ve döngüden sonraki ilk satıra devam eder.
- Önemli bir nokta, break sadece içinde bulunduğu en yakın döngüyü sonlandırır.





# continue

- Bu deyim döngünün o anki iterasyonunu sonlandırıp, döngü başına geri dönmesini sağlar.
- Yani continue çalıştığında, döngü bloğu içinde o noktadan sonraki kalan kodlar atlanır ve doğrudan bir sonraki döngü turuna geçilir.



# Sonsuz Döngülerden Kaçınma

- Sonsuz döngüler bilgisayarınızı kilitleyebilir veya programınızın yanıt vermez hale gelmesine sebep olabilir.
- Bu nedenle sonsuz döngülerden kaçınmak önemlidir.

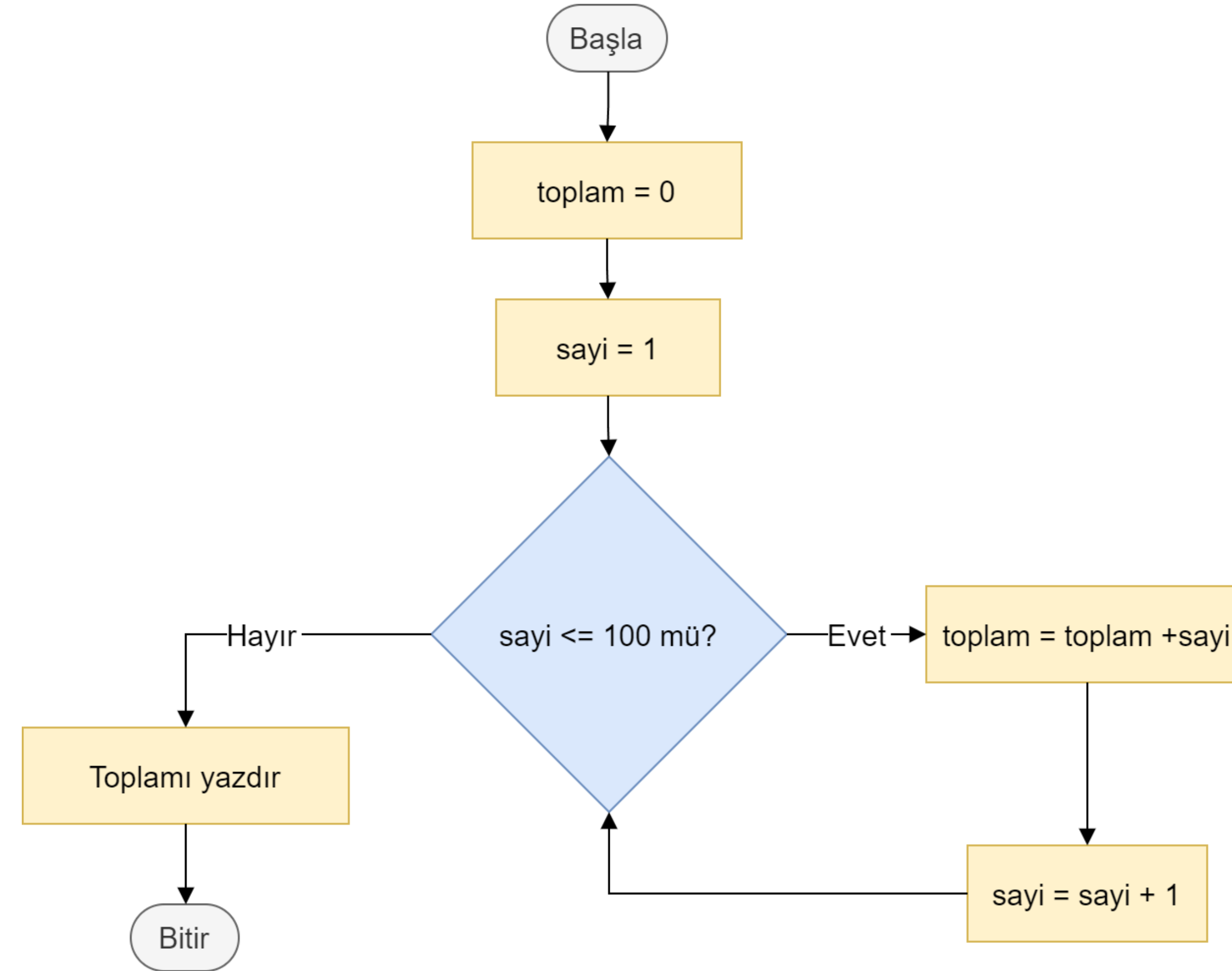
Döngü Koşulunu Doğru Güncelleme

break ifadesi ile çıkış

Kullanıcı girdi ile sonlandırma

Maksimum iterasyon sınırı koyma

# Sonsuz Döngülerden Kaçınma



# İç İçe Döngüler



```
1  n = 5
2  for i in range(1, n+1):
3      for j in range(i):
4          print("*", end="")
5      print()
```

\*

\*\*

\*\*\*

\*\*\*\*

\*\*\*\*\*

# Teşekkürler

ZAFER CÖMERT  
Öğretim Üyesi