

# Hata Türleri



"Ferah ol... Hata bizim özümüzde var..."

0 references

```
class Program
```

0 references

```
static void Main(string[] args)
```

```
{  
--  
}
```

```
namespace hata_kontrol_mekanizmasi
```

```
{
```

0 references

```
class Program
```

```
{
```

0 references

```
static void Main(string[] args)
```

```
{  
--  
}
```

```
}
```

```
class Program
```

```
{
```

0 references

```
static void Main(string[] args)
```

```
{
```

```
int isim.soyisim;
```

```
}
```

```
}
```

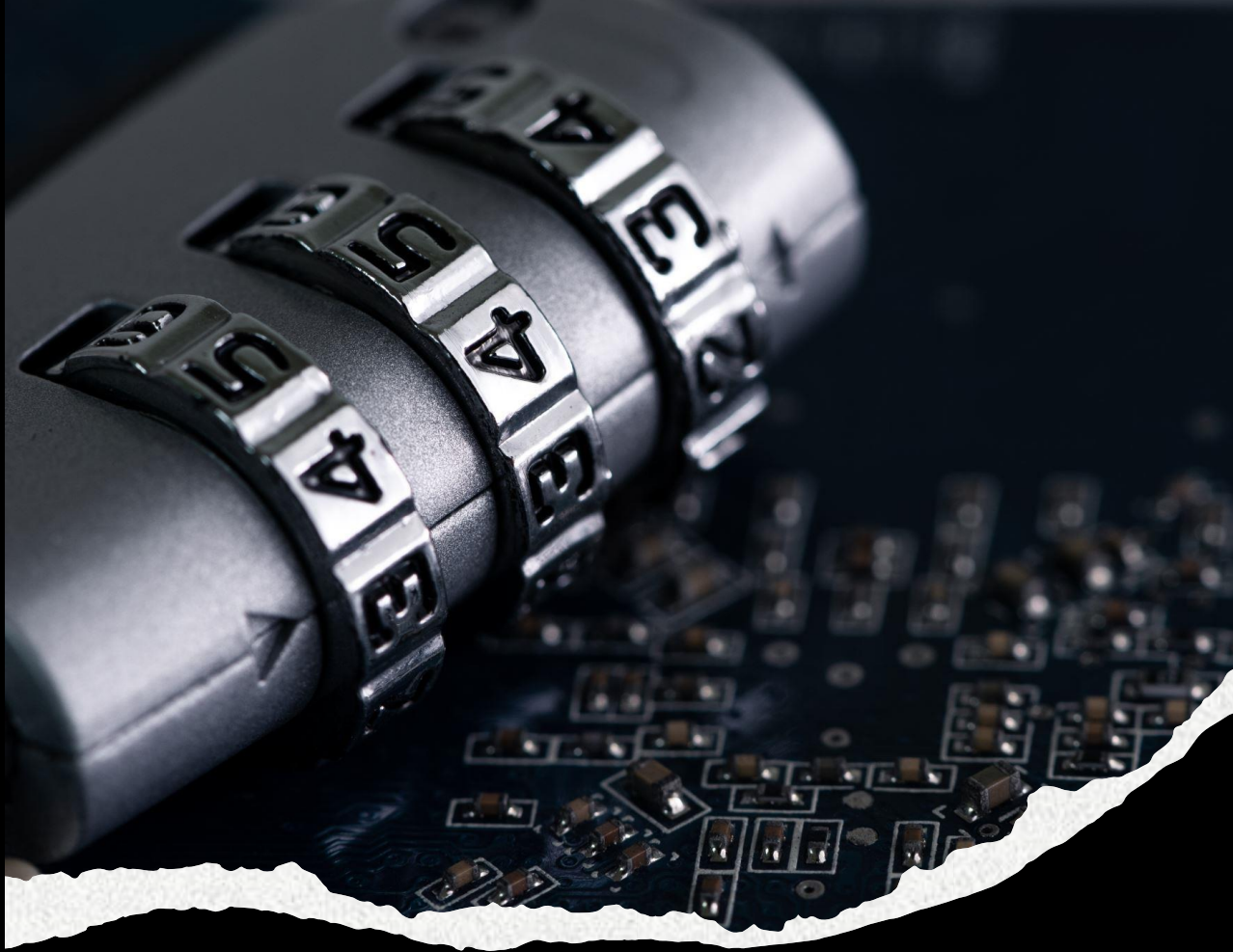
- Program dili kurallarına aykırı olan hatalardır.
- Özellikle gelişmiş editörler(VS) sayesinde söz dizimi hataları derlemeye gerek bile kalmaksızın fark edilebilmektedirler.
- Hatanın bulunduğu satır derleyici tarafından rapor edilir.
- O yüzden fark edilmesi ve çözülmesi ennnnn kolay hata türüdür.

## Syntax Error – Sözdizimi Hataları

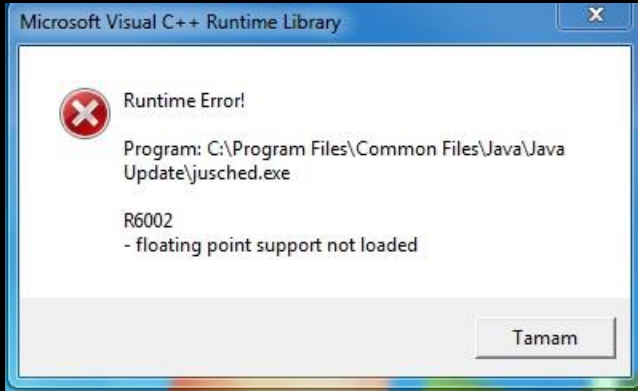


# Run-Time Error/Çalışma Zamanı Hataları

- Syntax'da bir problem yok,
- kodun semantiği kusursuz,
- çalışma zamanında patlamaya sebep veren hatalardır.



- Yazılım ayaktaiken ortaya ıkan bir takım aykırı durumlardan dolayı programın iřletim sistemi tarafından kesilmesiyle/sonlandırılmasıyla sonulanır.



- Çalışma zamanı hataları programın işleyişinin ortasında direkt kullanıcıyla temas edebilecek hatalardır.
- Ve hiçbir yazılımcı son kullanıcının hatayla karşılaşmasını istemez...
- Genellikle kültürden kültüre boyutu değişsede bir hatayla karşılaşan son kullanıcı derinden kulak kaşıdırabilir.
- Böyle bir durumda çalışma zamanında alınabilecek "olası" hataları yönetmemiz ve bir şekilde manipüle etmemiz gerekmektedir.



Çalışma zamanı hataları genellikle kullanılan mimaride öntanımlı mesajları verecektir.



Eğer ki alınan hatanın mimaride bir karşılığı yoksa işletim sistemi dili kullanılacak ve daha kompleks bir açıklamayla karşılaşılabacaktır.



OLMASI GEREKEN İSE ÇALIŞMA  
ZAMANI HATALARININ ÖNCEDEN  
TESPİT EDİLİP PROGRAMCI  
TARAFINDAN DAHA ANLAŞILABİLİR  
BİR ŞEKİLDE DÜZENLENMESİDİR.



TABİ BUNUN İÇİNDE ÇALIŞMA ZAMANI  
HATALARININ SAĞLAM BİR ÖNGÜRÜYLE  
TESPİT ÇALIŞMASI YAPILMASI  
GEREKMEKTEDİR.





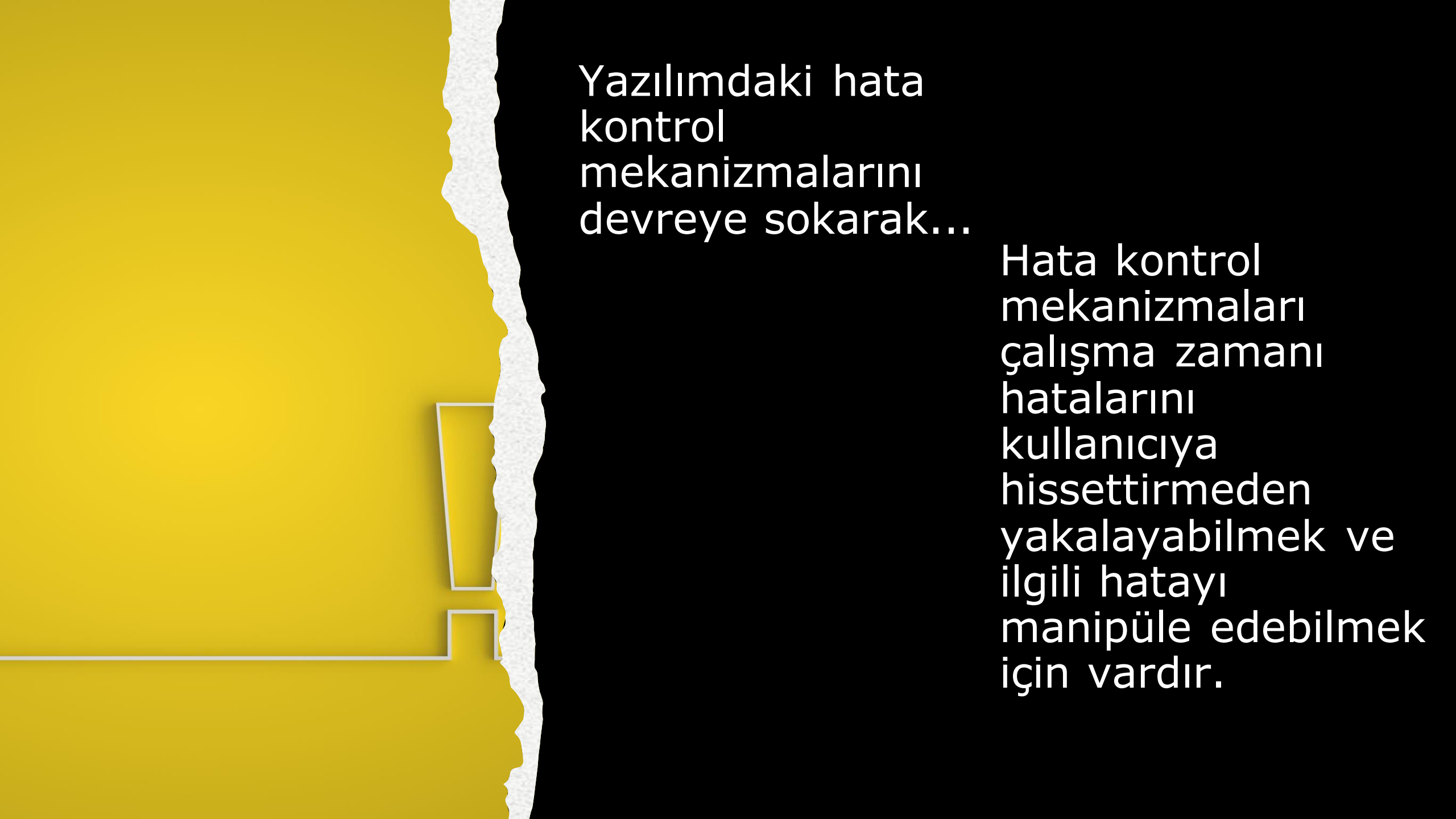
- Çalışma zamanı hatalarını önceden kestirmek oldukça zor olabilmektedir.
- Bu hataları öngörebilmek genellikle testerların işidir.
- Uygulama mümkün mertebe test edilerek çalışma zamanı hataları tespit edilmeli ve programcı tarafından tanımlanmalıdır.
- Tüm bunlara rağmen gözden kaçan durumlar olması oldukça normaldir. Dolayısıyla bir programın gelişimi sahadaki belli bir sürece bağlıdır.
- Bu süreçten toplanan loglar ve son kullanıcı dönütleri değerlendirilerek çalışma zamanı hataları tespit edilip arındırılabilir.



- Tespit edilen çalışma zamanlarının manipölasyonunun oldukça önemli olduğunu söyledik.

Peki bunu nasıl yapacağız?





Yazılımdaki hata  
kontrol  
mekanizmalarını  
devreye sokarak...

Hata kontrol  
mekanizmaları  
çalışma zamanı  
hatalarını  
kullanıcıya  
hissettirmeden  
yakalayabilmek ve  
ilgili hatayı  
manipüle edebilmek  
için vardır.



# Çalışma Zamanı Hata Durumları

- Olmayan bir dosyayı açmaya yahut üzerine yazmaya, okumaya vs. çalışmak
- Olmayan değer üzerinde işlem yapmaya çalışmak
- Uygun olmayan formatlarda çalışmak
- Veritabanı bağlantısının kopması



# Logic Error- Mantıksal Hatalar

Programın mantığında, akışında, algoritmasında, stratejisinde bir takım şeylerin yanlış hesaplanması, düşünülmesi, tasarlanması neticesinde alınan hatalardır.





Syntax'ta, kodun derlenmesinde  
ve hatta çalışma zamanında  
hata yoktur!

Kod çalışır. Sonuç verir.

Lakin sonuçlar hatalıdır.  
Beklenen sonuçlar elde  
edilemez!

Anlaşılan kodun mantığında ve  
hesapta bir yanlış vardır.





Mantıksal hatalar  
ancak test  
süreçlerinde yahut  
müşteri  
kullanımında tespit  
edilebilir.



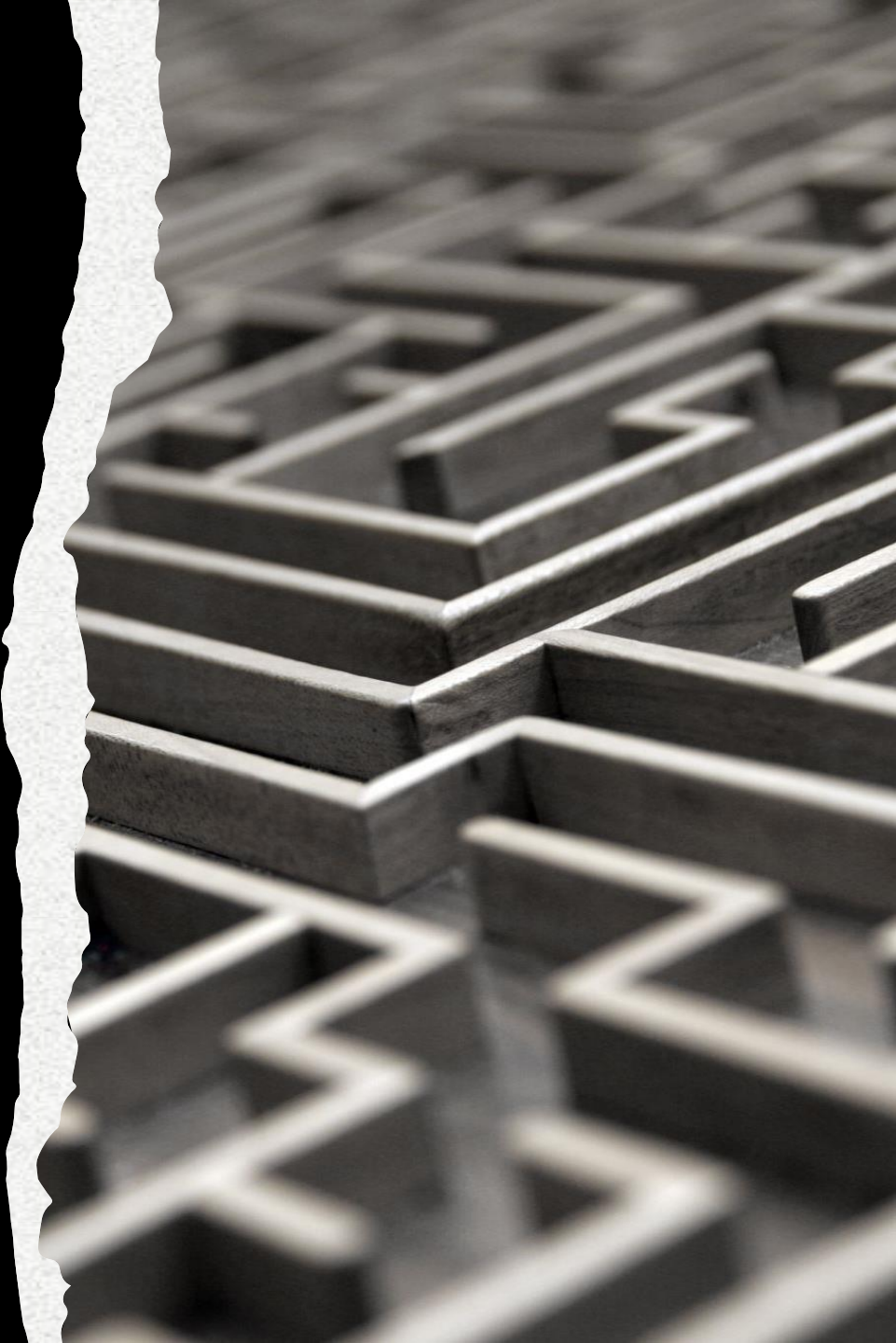


Bazen hesaplanması  
gereken bir deęerin  
eksik  
hesaplanmasıyla,  
bazen yanlış  
katsayının  
kullanılmasıyla,  
bazende mantıksal  
işlemdeki yapılan bir  
hatayla ortaya  
çıkabilir.

**Günlük hayattaki  
karşılığı 'Bug' dur.**



Tespiti çok zor olduđu için hata türleri arasında en tehlikeli hatadır.



Mantıksal hatalarda bazen tek çözüm  
debug'dır.

