

İşaret ve İşaretli

[İşaret](#) ve [İşaretli](#) anahtar kelimeleri birbirlerine çok benzeselerde aynı şeyi ifade etmezler.

Ortak tek yönleri, bir adrese işaret etmeleridir. Fakat işaret ettikleri adres ve ardında yatan konsept farklıdır.

İşaret neye işaret eder?

İşaret arabellekte bir konuma işaret eder. Daha doğrusu arabellekte önceden belirlenmesi zorunlu olan bir bayt-akımı alanına işaret eder.

Bu tür işaretlerle çalışmak risklidir. Eğer bellek yönetimi hakkında uzman değilsen, bu yöntemi kullanma. Bu işaretlerin işaret edecekleri adres giriş noktaları ve alanları işletim sisteminin çekirdeği tarafından sunulur.

Burada meydana gelen bir hata sonucu sistem uygulamayı zorunlu olarak sonlandırır ve bu tür hataları kod içinden önlemek mümkün değildir. Ancak bu tür işaretleri dikkatli kullanarak ve hata yapmamaya özen göstererek zorunlu sonlandırmaların önüne geçilebilir.

İşaretli neye işaret eder?

Basit iki tür işaret altyapısına sahiptir. Fiziksel işaretler ve sanal işarettir. Fiziksel işaretler yukarıda ifade edilen modeli kapsamaktadır.

[İşaretli](#) modeli ise sanal işaret altyapısıdır. Bu tür işaretler kendi içinde ikiye ayrılır:

Model	Anlamı
Türlü İşaret	Belli bir veri türüne göre işaret
Türsüz işaret	Her türden "türlü" işarete göre işaret

Teknik olarak bu işaret türü gerçek bir işaret değildir ve işletim sistemi çekirdeğiyle herhangi bir şekilde ilişkili değildir. Bu model daha çok klasör yolu gibi, [Basit Baytkod Motoru](#) tarafından yönetilen yol gösterici bir işaret konseptidir. Fakat kodda kullanırken gerçek bir işareten bir farkı yoktur.

Gerçek işarete kıyasla birçok avantajı vardır:

- Hata durumları [Basit Baytkod Motoru](#)'nun yönetiminde değerlendirilir
- Geçerliliği kesin olarak tespit edilebilir
- Uygulamayı illaki çöktürmez
- Hata yapma oranı çok düşüktür, çünkü türlü modeliyle önlem alınır
- İster 32Bit, ister 64Bit derlemede olsun, daima dört bayttan oluşan adrestir

İşaretli İşaret nedir?

İşaretli İşaret bir değişken tanımı olarak veya bir prosedür geri verme modeli olarak kullanılabilir. Burada aslında **İşaret** veri türüne sanal bir işaret belirlenir:

```
Prosedür Başlat ()
    İşaret veriAlanım Boyut 100 B
    Değişken İşaretli İşaret şurası
    şurası = veriAlanım
    ? ...
ProsedürSon
```

Normalde aynı işlemi boş bir işaret tanımıyla da yapabiliriz:

```
Prosedür Başlat ()
    İşaret veriAlanım Boyut 100 B
    Değişken İşaret şurası
    şurası = veriAlanım
    ? ...
ProsedürSon
```

Peki aradaki fark nedir?

Birinci örnekte **İşaretli İşaret** oluşturduk. Yani bu değişken sadece dört bayttır. Fakat yine de gerçek işarete ve onun temsil ettiği 100 bayt kapasiteye sahip bellek alanına işaret eder.

İkinci örnekte normal bir işaret oluşturduk. Bu işaret 32Bit derlemede yine dört bayttır, ancak 64Bit derlemede sekiz bayt olarak oluşturulur. Yani aslında gereksiz bellek alanı işgal etmiş olduk.

Bu iki temel fark dışında birinci modeldeki **İşaretli** yöntem daha güvenilirdir. Olası işaret hatalarını minimuma indirmek için en iyi çözümdür.

İşaret ve İşaretli değerleri farklı mıdır?

Evet farklıdır. Her ikisinde sayısal değerdir, fakat **İşaret**'in temsil ettiği veri sistem çekirdeği tarafından sunulur. Ancak burada sunulan sayısal değer sadece arabellek alanının geçerliliği sürece geçerlidir. Eğer alan tekrar kaldırılır ise, o halde işaret değeri geçersiz bir noktaya işaret eder ve hataya sebebiyet verir.

İşaretli ise **Basit Baytkod Motoru** tarafından oluşturulur ve yönetilir. Eğer işaret ettiği değer kaldırılrsa, söz konusu sayısal değer için Motor yine de **Boş** yanıtını verecek ve hata olasılığını ortadan kaldıracaktır.

From:

<https://basit.web.tr/dokuwiki/> - **Basit Yazılım Dili Dokümantasyonu**

Permanent link:

https://basit.web.tr/dokuwiki/doku.php?id=isaret_ve_isaretli

Last update: **11.12.2020 01:25**

