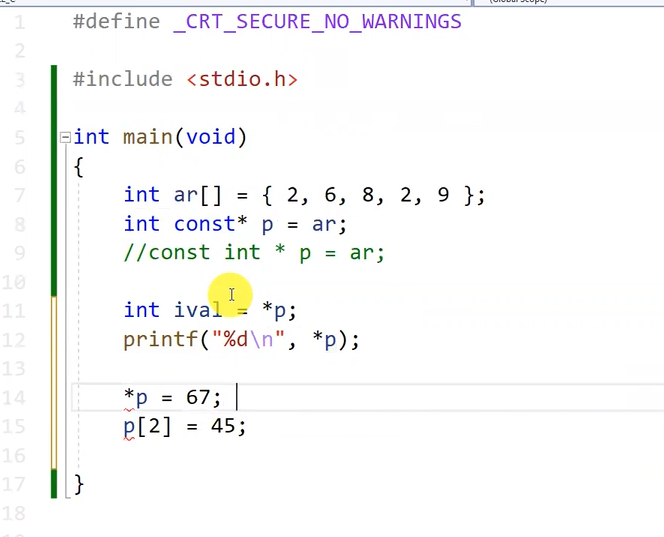
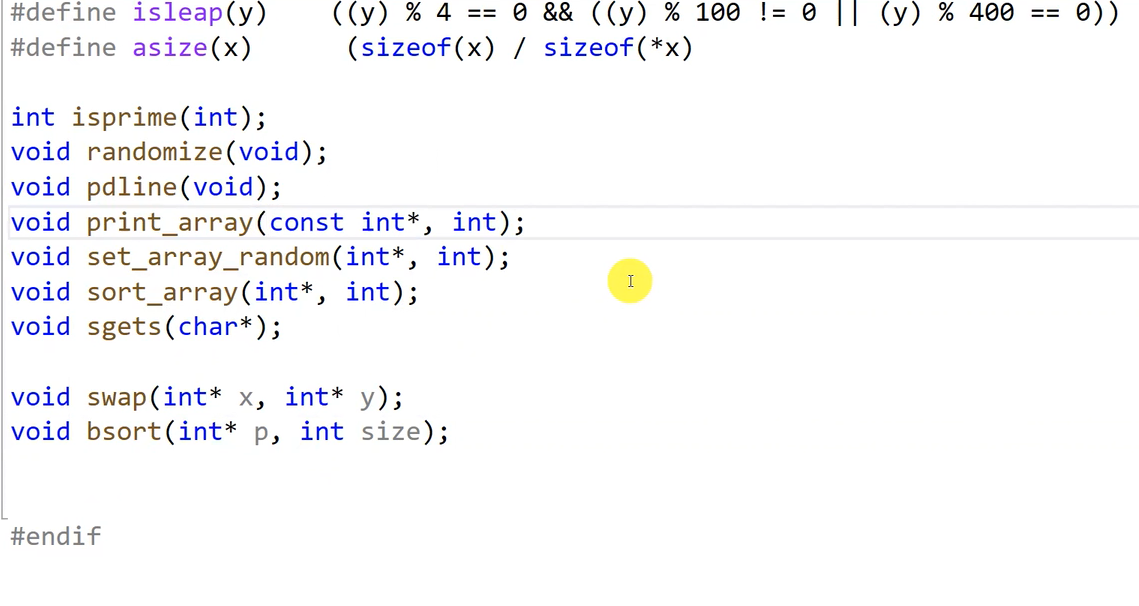
Ders 28

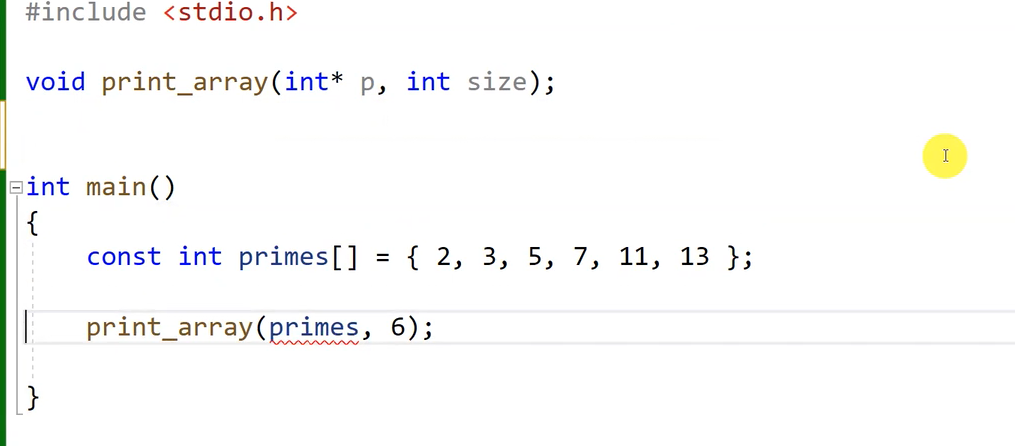
Const kullanımı



Son 2 ifade sentax hatası

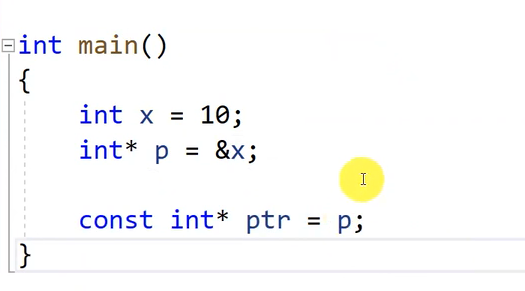


Print\_array const int var bunun nedeni dizinin elemanlarını değiştirmeden sadece ekrana yazdırmak olduğu için.

Örnek dönüşüm hatası

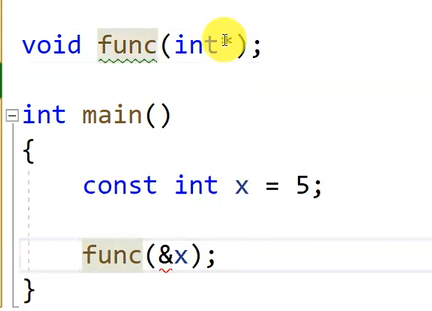
Const int int\* türüne dönüşürken hata olur uyarı verir vetehlikeli bir işlemdir. Değişmeyecekse fonksiyonda const int \* yazmalıyız.

Örnek hata var mı

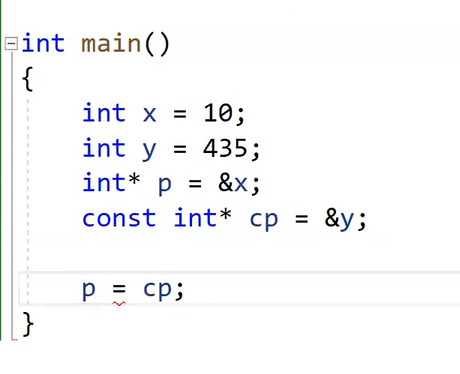


Bu kodda hata yok

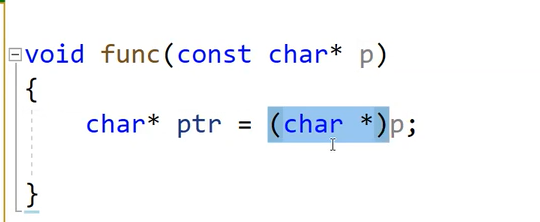
Tersi olsaydı syntax hatası değildi amakesinlikle yapmayın.



C++ hata c de uyarı verir yapmayın.

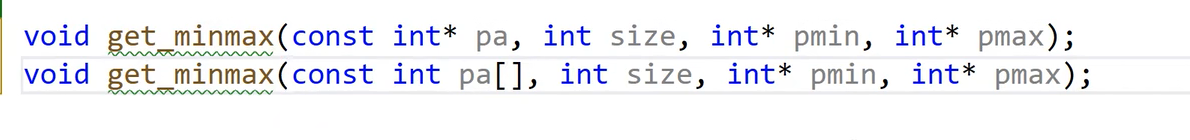


Burada const int\* dan int \* a dönüşüm var uyarı verir necatiye göre sakın yapmayın.



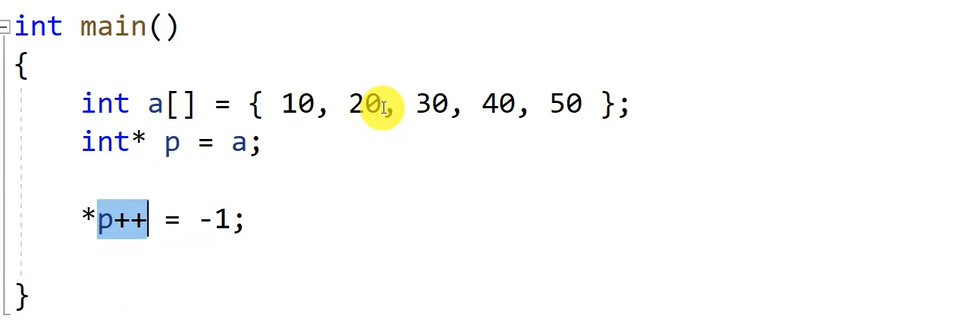
İsteyerek yaptığımızı göstermek için tür dönüştürme operatörü kullanmal ıyız.

Örnek konvensiyon



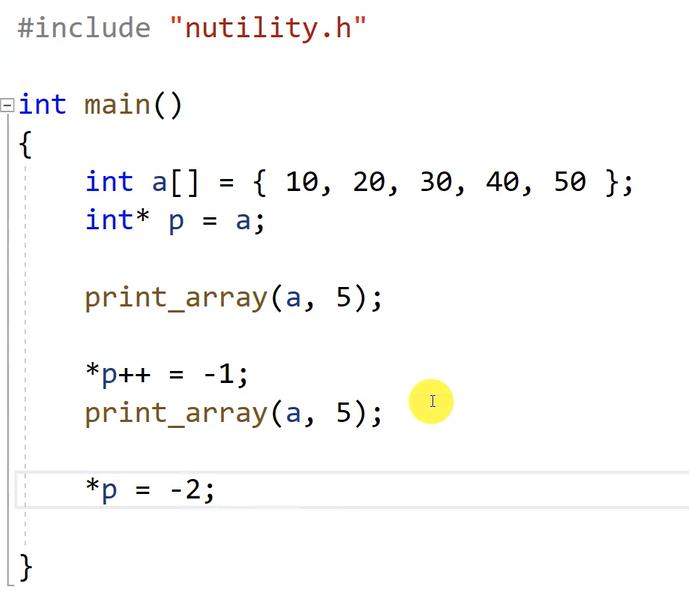
Ama ben üstekini yapıyorum.

Pointer idiom örnekleri



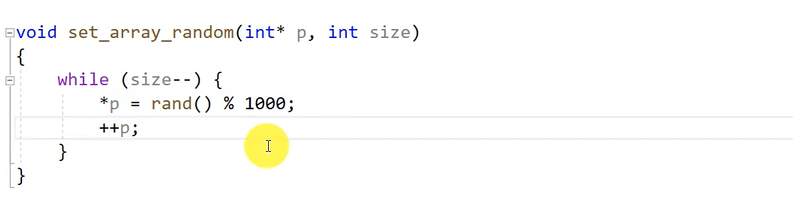
Geçerli ve l value 1. Elemana 1 değerini verdik

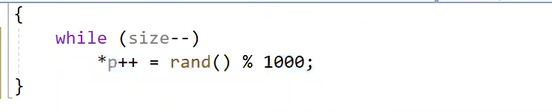
değerini gösterirdi



\*p++ adresi 1 arttırdı. \*p dizinin 2. Elemanını gösterir suan

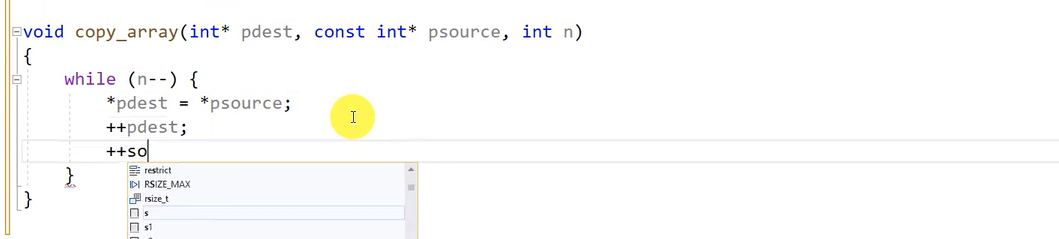
Örnek kullanım

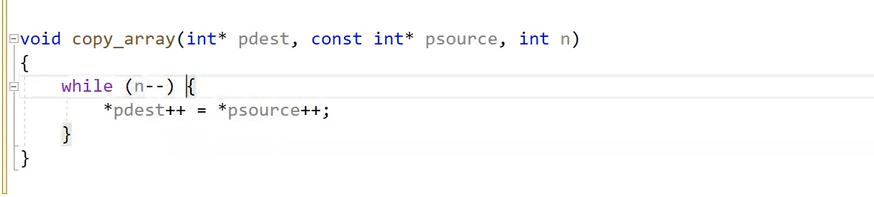




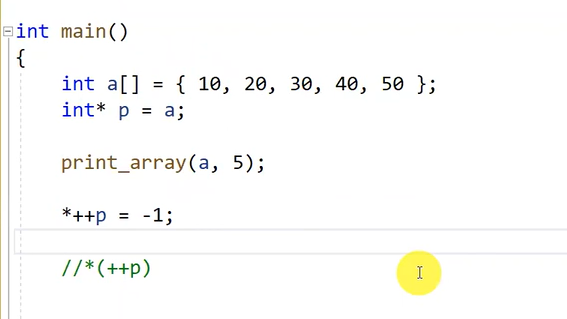
Es anlamlı

Copy array fonksiyonu çok güzel idiom örneği

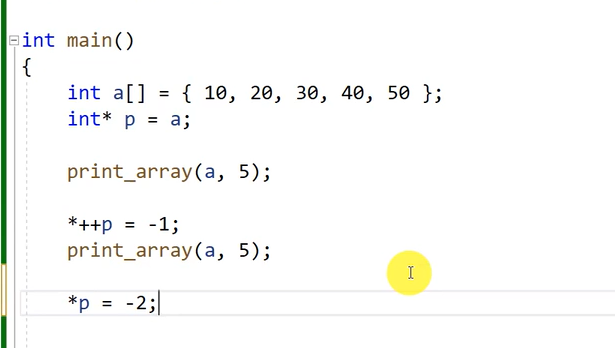




örnek

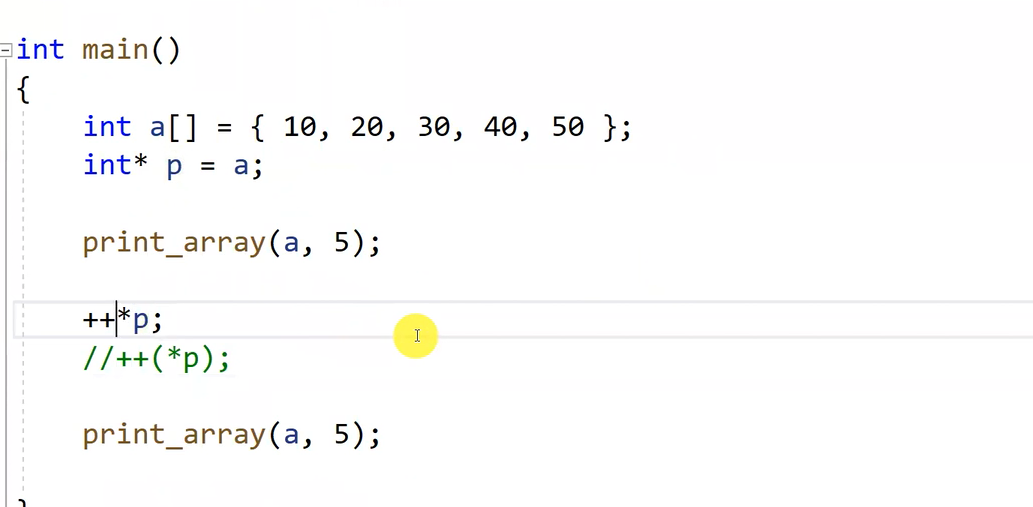


Hata yok dizinin 2 i elemanı -1 oldu..



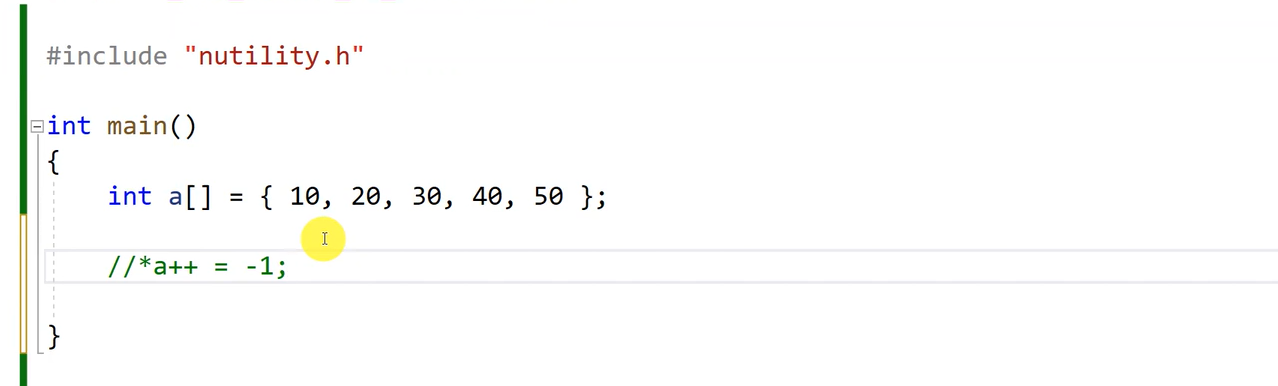
Buradada yine 2. Eleman -2 oldu.

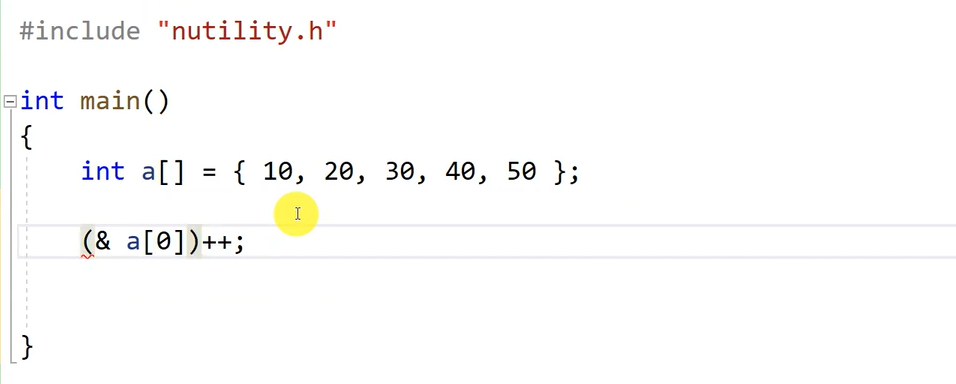
Örnek



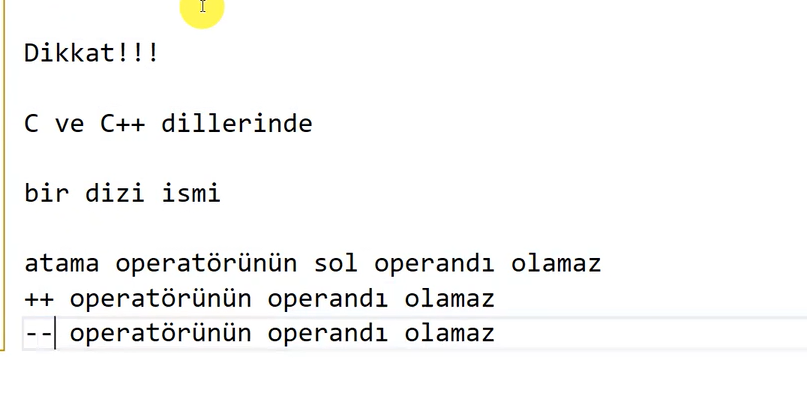
Burada 1. Eleman 1 artacaktı. Elemanın değeri 11 oldu \*p nesnesini 1 arrttırdı.

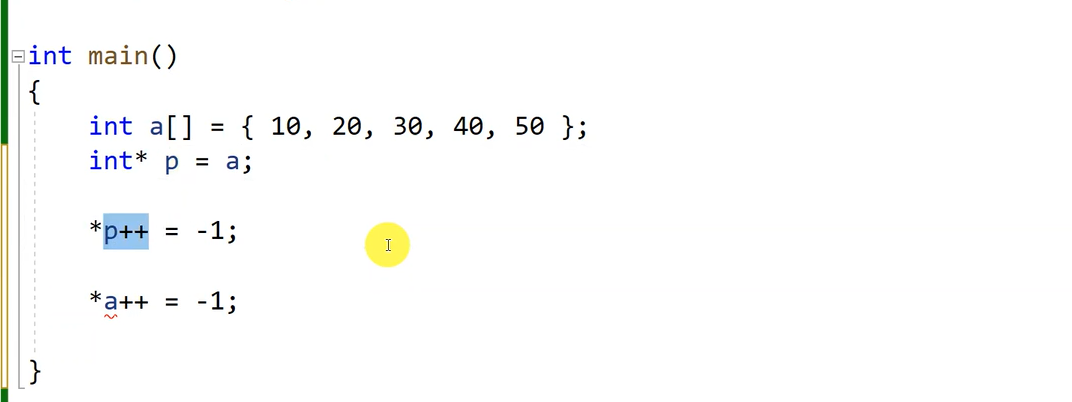
Şaşırtmalı soru hangi eleman -1 olur



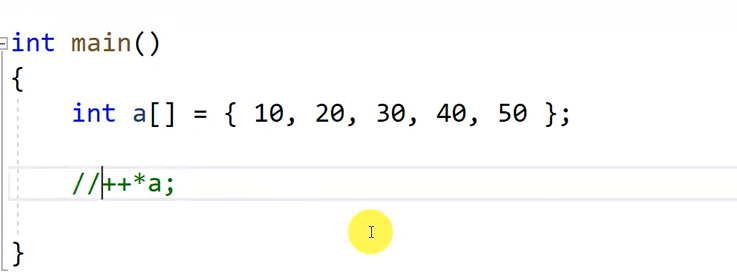


Aslında bu ifadeye dönüşüyor yukardaki ifadeler sentax hatası olur.



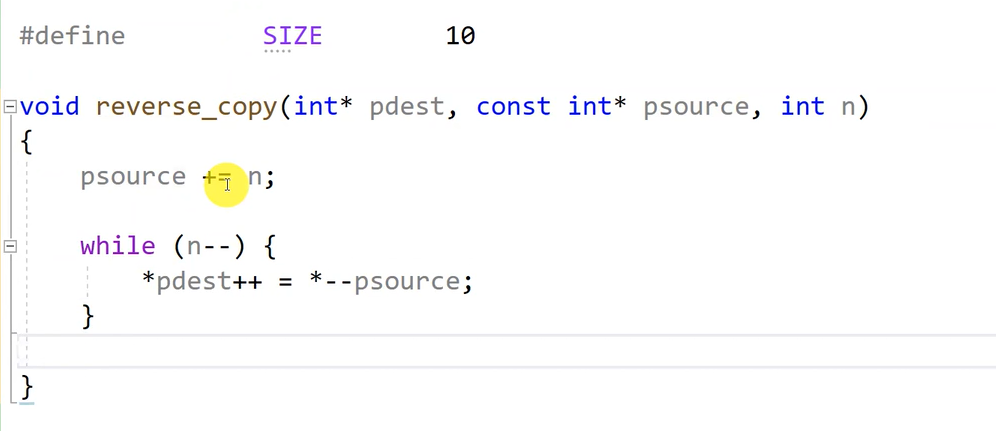


İlki geçerli



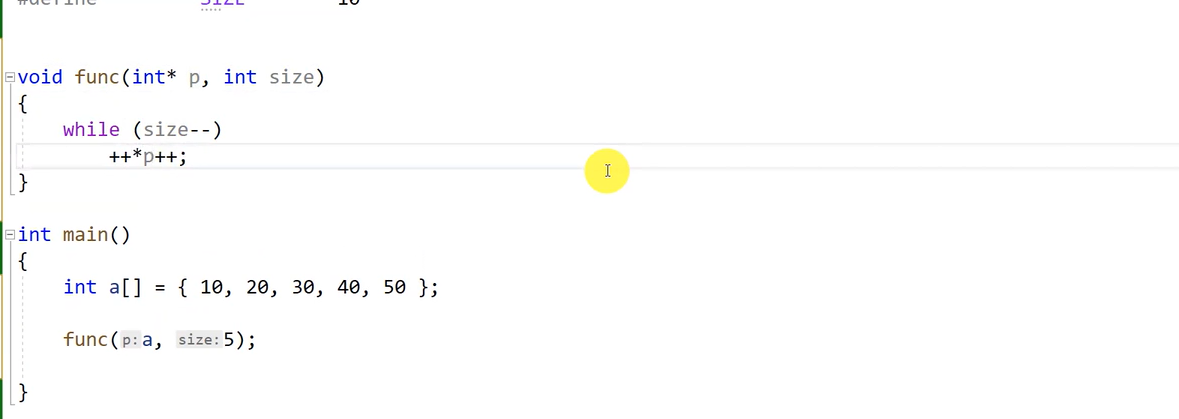
++\*a; geçerli kullanılabilir. Dizinin ilk elemanını 1 arttırır.

Çok önemli soru



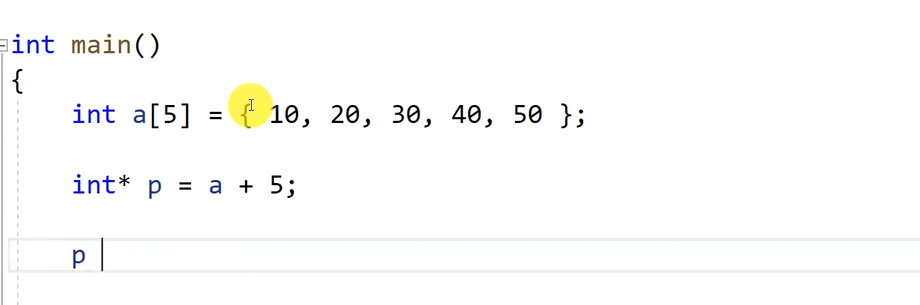
Dizinin bittiği yeri pointere atadım sonra –p yaparak 1 eksiği ile işleme girdim bu şekilde tasma olmadı.

Örnek



Adres 1 artsın gösterdiği değişkende 1 artsın ama kendinden başlayarak artacak.

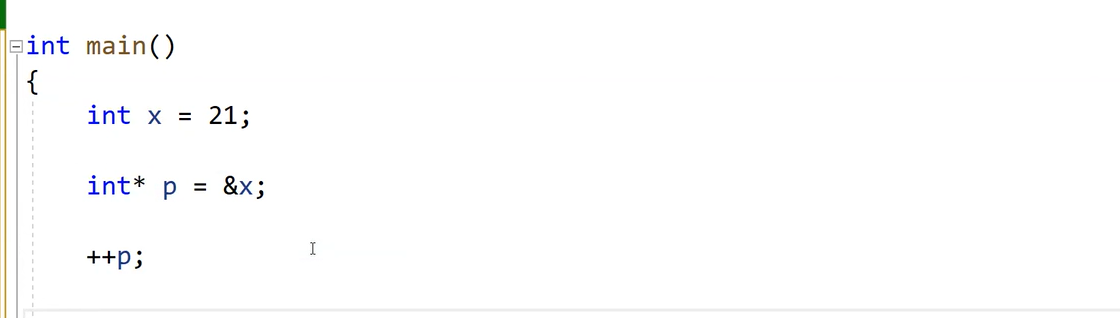
Örnek geçerli pointer



Dizinin son elemanından bir fazla adres değeri atadık ancak dizinin bittiği yerde geçerli bir ponter değeri oluşturuyor.

Kodlamada dizinin ilk elemanın adresine dizinin boyutunun eklendiği adrestir bir çok işlemde kullanıyoruz ancak içerik operatörünün operandı yapmıyoruz.

--\*p deseydim 1 ekseği yani diziyi sondan yazdırabiklirdim.



Tekil nesneler bir elemanlı dizi gibi işlenir.

Burada pointer değişkeni geçerlidir.