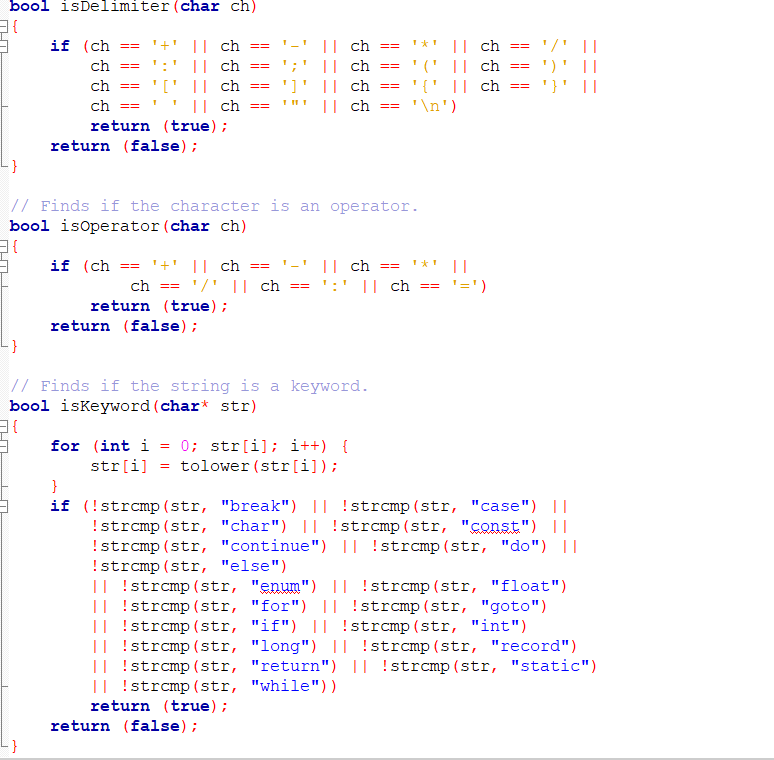


ConcatString fonksiyonu relloc ile dinamik bellek kullanarak ayrılan yerin boyunu arttırmak için kullandık .

isBracket fonksiyonu ile okunan karakterin köşeli ya da curly bracket içerip içermediğini return ediyor.

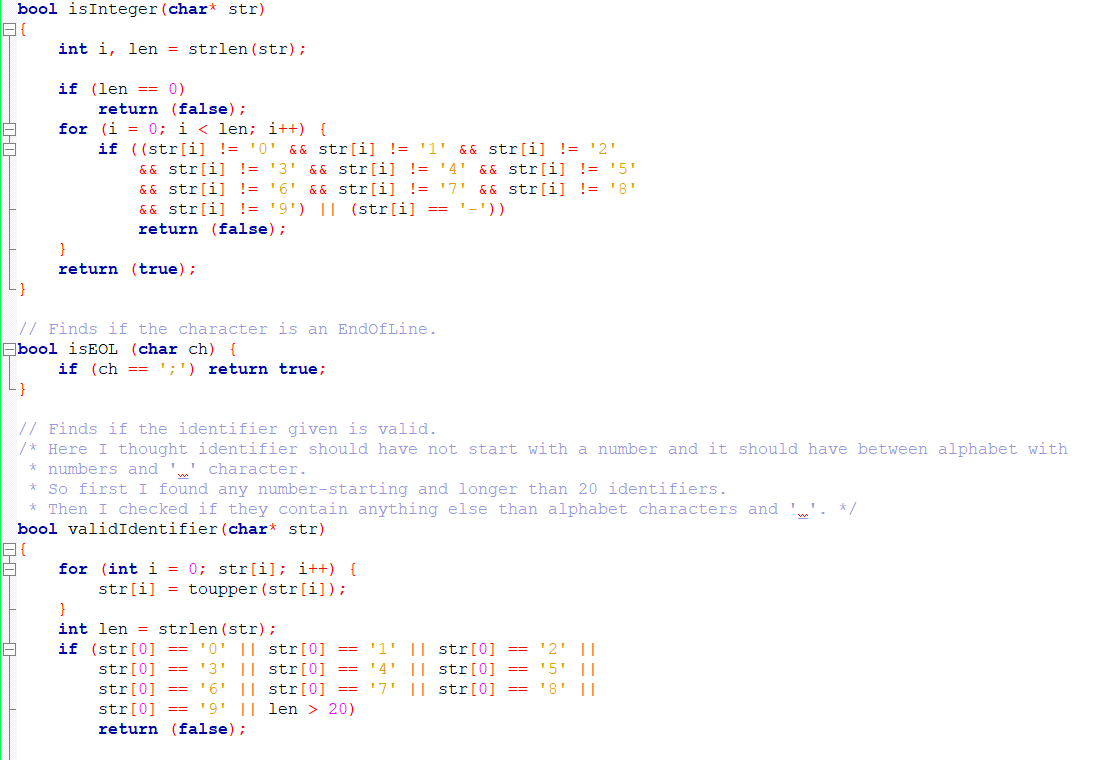
isDouble Quote ile “” bu karakterin olup olmadığını return ediyor



isDelimeter fonksiyonu yollanan karakterin içindeki noktalama işaretlerinin kontrolünü sağlamakta.

isOperator fonksiyonu yollanan karakterin içindeki matematiksel operatörlerin kontrolünü sağlıyor.

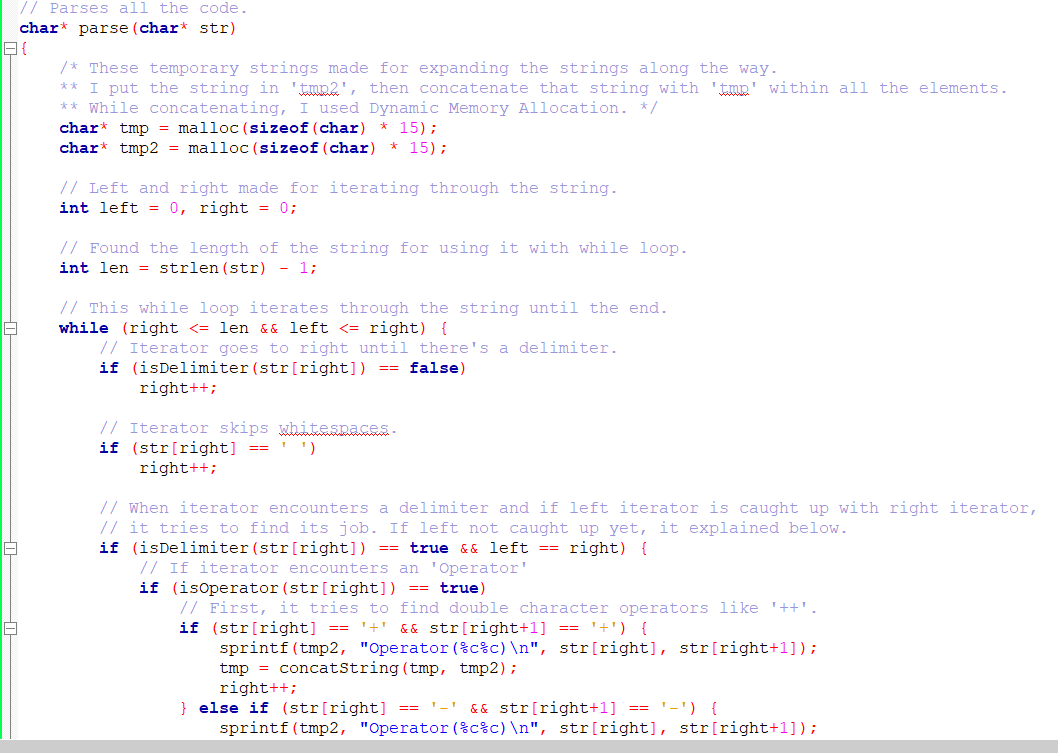
İsKeyword fonksiyonu yollanan şeyin keyword olup olmadığının kontrolünü sağlamakta.



isInteger fonksiyonu yollanan string’in keyword olup olmadığına bakmakta.

isEOL fonksiyonu satır sonuna gelip gelmediğimize bakmakta.

ValidIndentifier fonksiyonu verilen indentifier geçerli mi ona bakmakta uzunluğunun 20 den küçük olmasına ve sayı ile başlamaması gerektiği için onların kontrolünü sağlamakta.



Parse fonksiyonu ile üstteki fonksiyonlardan da yararlanarak kontrolleri yapıyoruz.

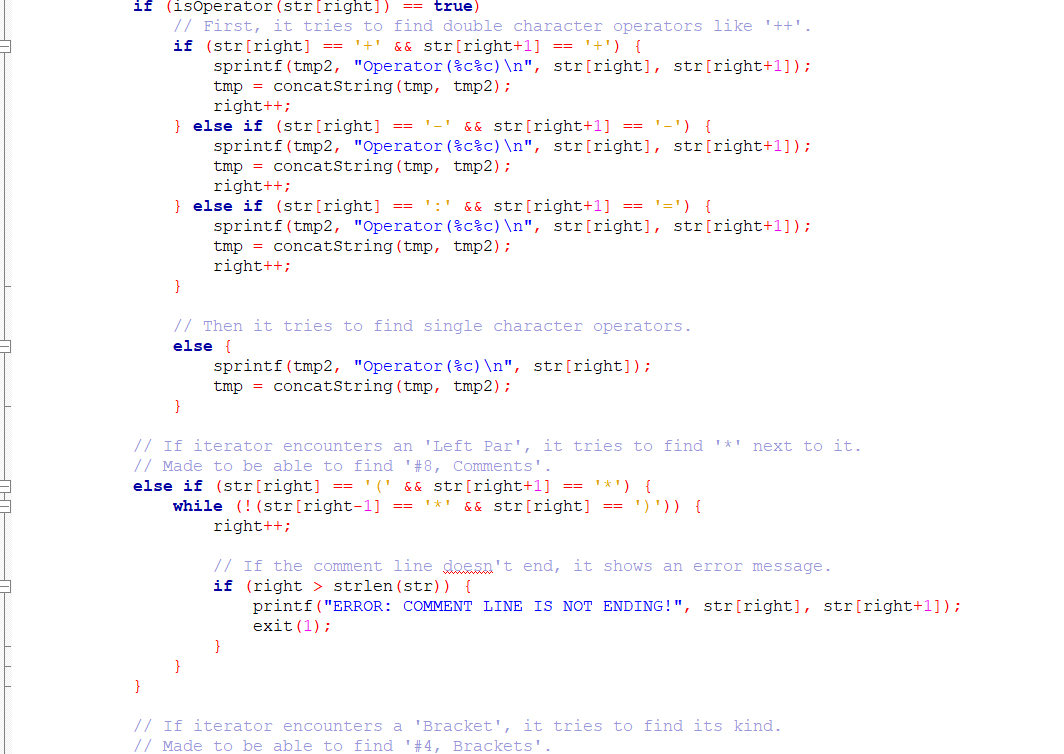
Geçici olarak 2 tane string değişken oluşturduk burada da dinamik bellek oluşturduk.

Len ile string’in uzunluğunu bulduk while döngüsü ile de baştan sona kontrol edeceğiz.

İterator sonuna denk gelene kadar sağa doğru gidecek şekilde dönen bir döngü var.

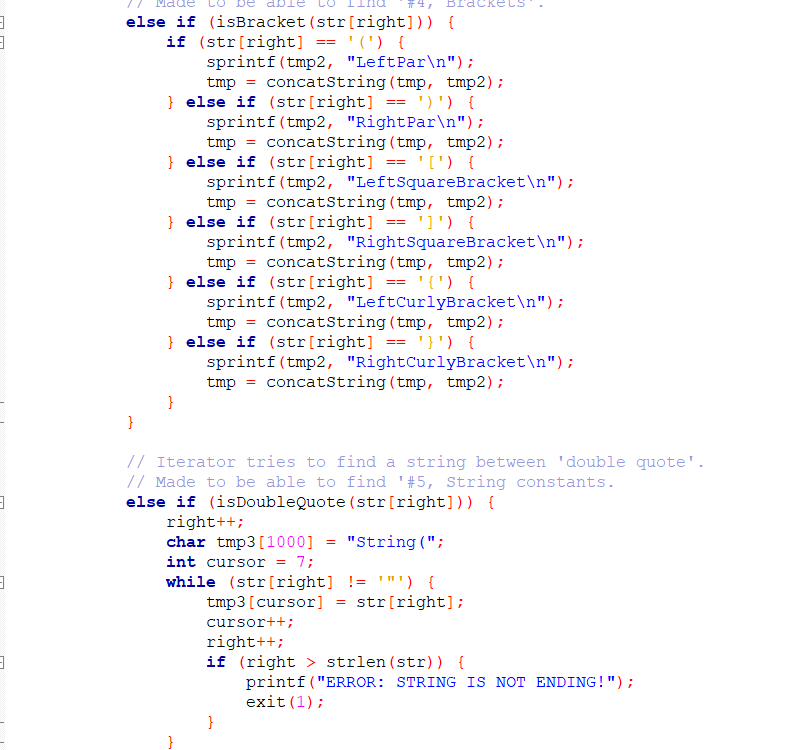
Alttaki if döngüsünde IsOperator fonksiyonundan çıkan sonuca göre yaptığımız kontroller değişmekte.

Alttaki else if durumları ile operatörleri kontrol ediyor.

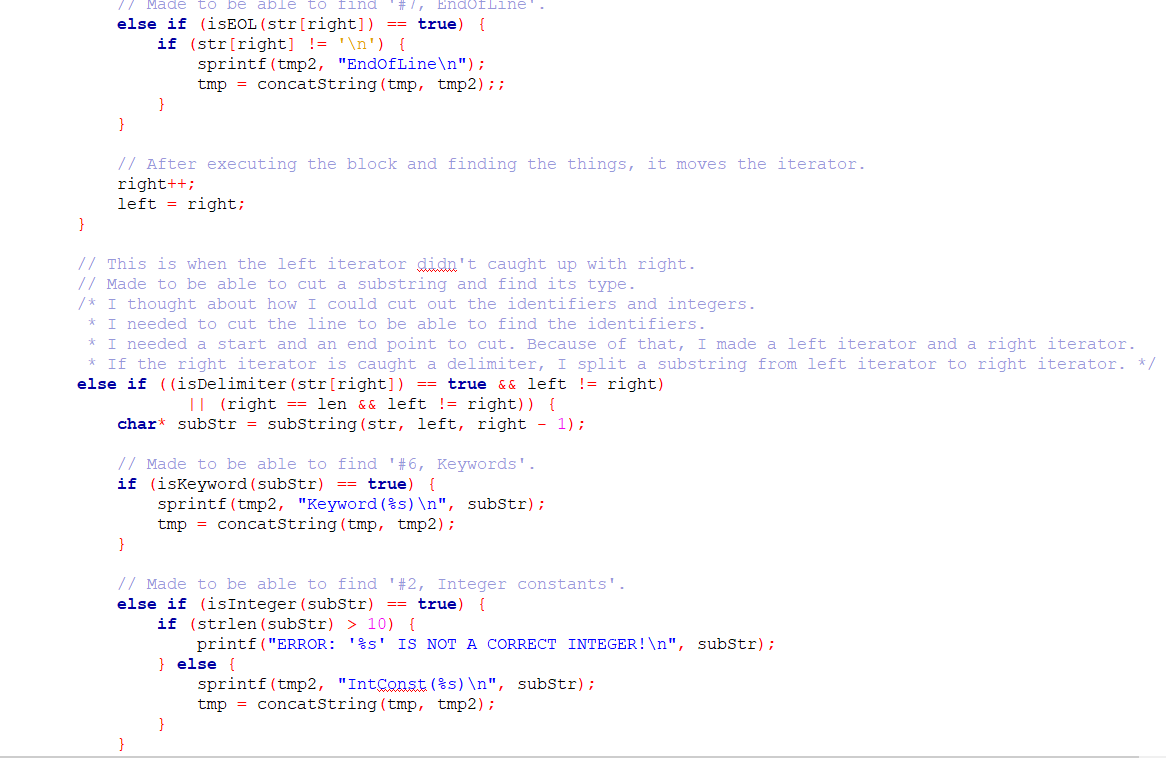


İteratör araken Left Par kısmında eğer ‘\*’ bu sembole denk gelirse yorum satırı’nı kontrol eder.

Yeni bir while döngümüz var burada bunun içinde ilk olarak satırın bitip bitmediğini kontrol ettik.



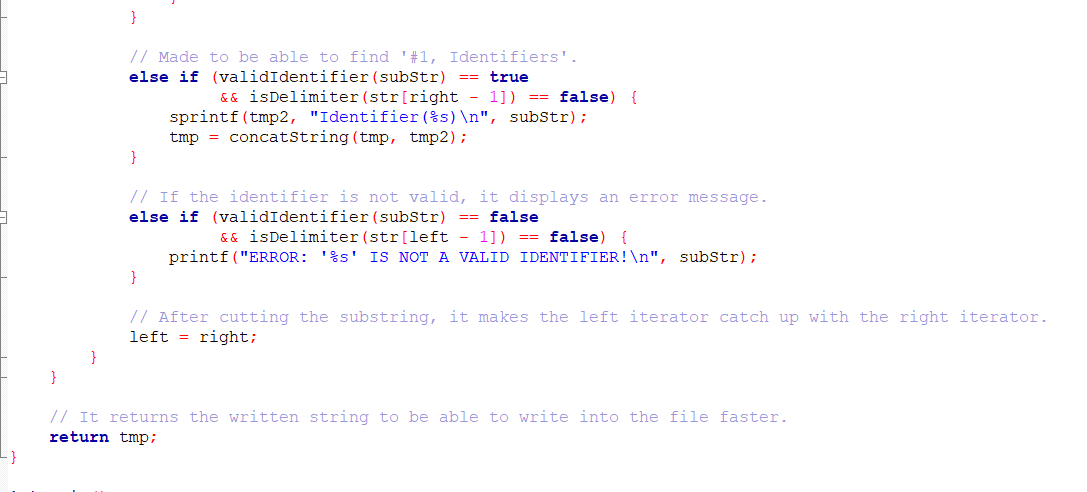
Burada ilk önce parantezleri kontrol ettik sonra altındaki kısımda ise , “” arasındaki stringleri bulmaya çalıştık .



En sonda çalışıp bir şeyler buldukça belirleyici’leri güncelledik . Onun altındaki kısımda ise ,

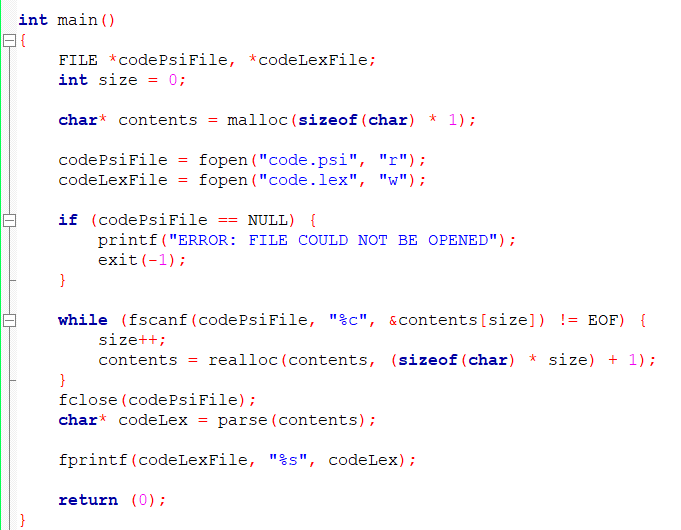
Eğer sol belirleyici sağdakini yakalayamazsa nasıl olacağını düşündük.

Ve bazı kontroller sonucu ekrana hata çıktısı yazdırdık.



En sonda ise bulduğumuz sonuca göre kontrol ettiğimiz geçici değişkende tuttuğumuz string’i return ettik.

Alt kısımda ise main fonksiyonu ve onun içinde yapılan işlemleri göreceksiniz.



Burada ise dosyayı açtık eğer açılırken bir problem olursa ekrana hata mesajı yazdırır.

Açıldıktan sonra içinden gereken şekilde karakterleri aldık en sonda ise işimiz bitince dosyayı yeniden kapattık.

Altta ise yazılan dizinler ve bunların ekran çıktılarını göreceksiniz.

