



**T.C.**  
**SAKARYA ÜNİVERSİTESİ**

**BİLGİSAYAR VE BİLİŞİM BİLİMLERİ FAKÜLTESİ**  
**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**  
**PROGRAMLAMA DİLLERİNİN PRENSİPLERİ ÖDEV RAPORU**

**ÖDEV BAŞLIĞI**

**G201210044 - Ali Said Saritemur**

**SAKARYA**

**Mart, 2022**

Programlama Dillerinin Prensipleri Dersi

# Operatör Sayacı

Ali Said Sarıtemur

<sup>a</sup> Ali Said Sarıtemur 2C

## Özet

Problem operatörlerin türlerine göre bir sınıflandırma yapıp bunları saymak ve saydığımız değerleri türleriyle birlikte çıktı olarak vermektir. Bunun için de regex'i kullandım. Bunun dışında uygulamayı yorum satırlarına duyarlı hale getirdim. Burada sadece üstünkörü bahsettiğim bu durumlar Geliştirilen yazılım kısmında detaylıca açıklanıyor bu yüzden maalesef ki özet kısmına daha fazla yapabileceğim bir eklenti bulunmamakta.

© 2017 Sakarya Üniversitesi.

Bu rapor benim özgün çalışmamdır. Faydalanmış olduğum kaynakları içerisinde belirttim. Her hangi bir kopya işleminde sorumluluk bana aittir.

Anahtar Kelimeler: Regex, Operatör, pattern.

## 1. GELİŞTİRİLEN YAZILIM

Ödevimde 5 class bulunmakta bunlar Main, Lexical, Operator, DosyaOkuma ve Sayac. İlk olarak DosyaOkuma classı BufferedReader ile içine atılmış olan dosya yolundaki dosyanın okunmasını ve bir string değişkeninin döndürülmesini sağlıyor. Sayac class'ı ise aslında ödevin en önemli yeri olan okunmuş olan dosyanın içindeki operatör ve operandların sayılması işlemini gerçekleştiriyor. Bunu yaparken de ödevde istenen durumların her biri için bir fonksiyon tanımlı. Bu fonksiyonlardaki Patern ve Matcher değişkenleri oluşturup bunlarla matcher.find() komutu ve while döngüsü vasıtasıyla regex komutundaki değerlerin sayımını sağladım. Bunun dışında yine Sayac classının içinde tanımladığım yorumSatiriTemizle fonksiyonuyla gönderilen string'in içindeki yorum satırları temizlenip yeni bir string değer döndürülür. Operator classında constructor içinde dosyaOkuma ve Sayac classları bir arada kullanılıp kendi içindeki değişken türlerine değer atanır. Mainde ise Operator classı tanımlanıp içine bir konup atanır ve get metodu ile değişkenler çağırılıp çıktı alınır. Bunun dışında Sayısal, Mantıksal, İlişkisel operatörleri sistemde belirtilmiş operatörlere göre saydırtım. İkili ve tekli operatörlerde ise tekli operatör olarak sadece ++ ve -- operatörlerini aldım, ikili operatörlerde de +, -, \*, = operatörlerini aldım. Toplam operand sayısını hesaplarken de toplam operatör sayısını ikiyle çarpıp çıkan sonuçtan tekli operatör sayısını çıkarttım. Lexical class'ı ise kullanıcının doldurması gereken class olarak tanımlandı. Bide dist dosyasının içine runnable JAR formatında dosyayı export ettim.

## 2. ÇIKTILAR

```
Operator Bilgisi:
    İkili Operator Sayisi: 6
    Tekli Operator Sayisi: 1
    Sayısal Operator Sayisi: 7
    İlişkisel Operator Sayisi: 3
    Mantıksal Operator Sayisi: 2
Operand Bilgisi:
    Toplam Operand Sayisi: 23
```

Ödevin örnek dosyasının çıktısı.

## 3. SONUÇ

Bu ödevin bize sağladığı en açık fayda tabii ki de regexin nasıl kullandığını öğretmesi. Çünkü bu ödev gibi gerçek hayatta da karşılaşma olasılığımızın olduğu sayma işlemlerinde bizleri onlarca if else koşullarından oluşan karışık kodlar yerine tek satır ile halledebileceğimiz bir kod dizini sunuyor.