



T.C.
SAKARYA ÜNİVERSİTESİ

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

PDP ÖDEV 1

B191210044 - Burak Kozluca

SAKARYA

Nisan, 2022

Programlama Dillerinin Prensipleri Dersi

Dosya Okuma ve Operatör Sayısı Bulma

Burak KOZLUCA

B191210044 1/C

Özet

Yazmış olduğum Java programı “Deneme.java” isimli Java dosyasını okuyarak içerisinde bulunan operatör sayısını (tekli operatör, ikili operatör, sayısal operatör, ilişkisel operatör, mantıksal operatör) ve operand sayısını regex yardımıyla bulmaktadır.

© 2022 Sakarya Üniversitesi.

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

Anahtar Kelimeler: Regex, Dosya okuma , Class,

1. GELİŞTİRİLEN YAZILIM

İlk olarak Program.java (Çalıştırılabilir sınıf) ve Lexical.java (operatör ve operand bilgisini tutacak sınıf) dosyalarını oluşturduk. Program.java ve Lexical.java dosyalarına projede kullanmak için gerekli olan kütüphaneleri ekledik. Lexical.java dosyamızın içinde gerekli regex tanımlamalarını yaptık. Program.java dosyamızda ise main class'ının içinde ilk olarak dosyayı satır satır okuma işlemini yaptık ve try-catch yapısı sayesinde dosya okunamazsa kullanıcıya hata fırlattık. Ardından her satır için Lexical.java dosyasında tanımladığımız regex ifadeler sayesinde satırlardaki operatör sayılarını çeşitlerine göre Matcher yapısıyla yakaladık. Akabinde her operatör çeşidi için ayrı while döngüsü kullandık. Ve her döngü içerisinde okuduğumuz satırı diziye atıp bu dizide yorum satırı kullanılmış mı onun kontrolünü yaptık. Yorum satırı kullanılıyorsa o satır içindeki operatörleri almıyoruz. Eğer satır yorum satırı ifadesiyle başlamıyorsa o satırda bulduğumuz operatör sayısını array liste ekledik. Bu işlemi her operatör çeşidi için ayrı ayrı yapıyoruz. Yani her operatör sayısı ayrı array listte tutuluyor. Son olarak ekrana yazdırılacak bilgileri ödev dosyasındaki formata uygun bir şekilde ekrana yazdırdık.

2. ÇIKTILAR

```
5<x
++y;
this.x = x;
/*
 *
 * x*x + y*y // işlemi x > 0 ve y > 0
 */
return Math.Log(((double)x)/y);
@Override
if(obj != null && obj instanceof Deneme) {
return ((Deneme)obj).x == this.x && ((Deneme)obj).y == this.y;
```

Yukarıdaki satırların okunması sonucunda programımız aşağıdaki çıktıyı vermektedir. Yorum satırlarındaki operatörleri dahil etmemektedir.

Operatör Bilgisi:

```
Tekli Operatör Sayısı: 1
İkili Operatör Sayısı: 8
Sayısal Operatör Sayısı: 3
İlişkisel Operatör Sayısı: 4
Mantıksal Operatör Sayısı: 2
```

Operand Bilgisi:

```
Toplam Operand Sayısı: 17
```

3. SONUÇ

Bu ödevle birlikte regex konusuna hakim oldum. Ayrıca bazı durumlarda daha pratik düşünerek çözüm yolları aramam gerektiğini öğrendim.

Referanslar

- [1] <https://regexr.com/>
- [2] <https://regexlearn.com/>