

Doğu Akdeniz Üniversitesi

BLGM318 Programlama Dilleri Prensipleri İlkbahar 2024-2025

Ödev #2

Sözcük analizcisi uygulaması (Lexical Analyzer)

İki kişilik gruplar halinde yapılacak. Tek kişinin yapması da mümkün.

Yapılacak İş:

Python dilinde yazılmış, tamsayıları (örnek: 4, 25, -42 vs.), kesirli sayıları (örnek: 3/5, 23/9 vs.), anahtar sözcükler **while**, **for**, **repeat**, **and**, **or**, **not** ve kullanıcı tanımlı isimleri (örnek: **x**, **y**, **z**, **e345**, **kapi_mandali** vs.) tanıyan sözcük analizcisi.

Sözcük analizciniz (**lex** diye bir fonksiyon) her çağrıldığında girdiyi dosyadan okumalı ve bulunduğu sözcüğü (lexeme), sözcüğe denk gelen jetonu (token), jeton eğer ISIM ise ismin sembol tablosundaki pozisyonunu (index), jeton TAMSAYI ise okunan tamsayının değerini (integer olarak), jeton KESIR ise kesirin pay ve paydasını ek bilgi olarak döndürmelidir.

Ana program önce girdi dosyasının adını okumalı ve aşağıdaki menü'nün gösterildiği **döngüye** girmelidir:

- 1) Sözcük analizcisini çağır ve döndürdüğü değeri ekranda göster
- 2) Sembol tablosunun güncel halini göster
- 3) Çıkış

1) numaralı opsiyon bir sonraki jetonu ve diğer gerekli bilgileri (örneğin sembol tablosunun içine işaretçi) almak için sözcük analizcisi fonksiyonunu çağırmalı. Dosyadaki boşluklar atlanmalı. Kullanıcı tarafından tanımlanan isimler, ilk görülmede sembol tablosuna konulmalı. Ondan sonraki görülmelerde, sembol tablosunda varolan girdi bulunup döndürülmeli. **Tanınmayan bir girdi bulunduğu HATA jetonu döndürülmeli.**

Sözcük analizcisi tarafından döndürülmesi gerek JETONLAR ve ek bilgiler şöyledir:

Sözcük (lexeme)	Jeton (Token)	Ek bilgi
Tam sayı tanımına uyan dizi	TAMSAYI	Tamsayının nümerik değeri
Kesirli sayı tanımına uyan dizi	KESIR	Pay ve paydanın nümerik değerleri
“while”	WHILE	
“for”	FOR	
“repeat”	REPEAT	
“and”	AND	
“or”	OR	
“not”	NOT	

Kullanıcı tarafından tanımlı isim (örnek “x”, “fatma tezye”, “y6” vs.)	ISIM	Sembol tablosunda index (ismin var olduğu veya konulduğu yer)
Yukardaki jeton türleri dışında bir girdi	HATA	

Sembol tablosunu anahtar sözcüklerle ilkleysin. Bu durumda “while” sıfır pozisyonuna, “for” bir pozisyonuna vs. yerleşmeli. Yani sembol tablosunda ilk 6 pozisyon program başlarken dolmuş olacak.

Bir isim bulunduğunda eğer isim hali hazırda sembol tablosunda var ise, sembol tablosundaki yeri (index’i) geri gönderilecek. Aksi halde önce isim tabloya konulacak, sonra yeri geri gönderilecek.

Örnek dosya içeriği (bu sadece BİR örnek! Programınız herhangi bir dosya içeriği ile çalışabilmeli!)

x 45 5/4 -33 size and y1234 45asdas not size x

Sözcük analizcisi bu dosya için aşağıdaki bilgiyi (her çağrıldığında bir tanesini!) döndürmeli:

Sözcük: “x”
Jeton: ISIM
Sembol tablosu: 6

Sözcük: “45”
Jeton: TAMSAYI
Değer: 45 (integer olarak!)

Sözcük: “5/4”
Jeton: KESIR
Pay: 5
Payda: 4

Sözcük: “-33”
Jeton: TAMSAYI
Değer: -33 (integer olarak!)

Sözcük: “size”
Jeton: ISIM
Sembol tablosu: 7

Sözcük: “and”
Jeton: AND

Sözcük: “y1234”
Jeton: ISIM
Sembol tablosu: 8

Sözcük: “45asdas”
Jeton: HATA

Sözcük: “not”
Jeton: NOT

Sözcük: “size”
Jeton: ISIM
Sembol tablosu: 7

Sözcük: “x”
Jeton: ISIM
Sembol tablosu: 6

Bu neticeler sözcük analizcisi çağrıldıkça ekrana yazılmalı.

2) numaralı opsiyon seçildiğinde sembol tablosunun güncel hali ekranda gösterilmeli (girdilerin doğru yapıldığını denetlemek için).

3) numaralı opsiyon seçildiğinde program sonlanmalı

Notlandırma: Projenin kendisi, ve ortakların proje hakkındaki bilgileri ayrı ayrı notlandırılacaktır.

Verilecekler (Teams ve Moodle üstünden):

1. Problemi ve çözümünü anlatan rapor (pdf olarak). Raporun ana kısmında problemin ne olduğu ve kodun önemli parçalarının çalışması anlatılacak.
 - a. Ek 1’de program kodu
 - b. Ek 2’de en az 15 girdi bulunan girdi dosyası
 - c. Ek 3’de girdi dosyasının çalıştırılması ile elde edilen çıktılar (ekran görüntüleri)
2. Sözcük analizcisi kodu (Python .py dosyası)
3. Girdi dosyası (.txt uzantılı)