

LAPORAN AKHIR PRAKTIKUM

Mata Praktikum : Teknik Kompilasi

Kelas : 4IA27

Praktikum ke- : 1

Tanggal : 13 Oktober 2024

Materi : Tahapan Kompilasi

NPM : 51421476

Nama : Theodore Gabbelambok Situmorang

Ketua Asisten : Dimas Renaldy

Paraf Asisten :

Nama Asisten :

Jumlah Lembar : 5 Lembar

LABORATORIUM TEKNIK INFORMATIKA

UNIVERSITAS GUNADARMA

2023

LAPORAN AKHIR 1

```
import re

JENIS_TOKEN = {
    'KATA_KUNCI': r'\b(if|else|for|while|return)\b',
    'IDENTIFIKATOR': r'\b[A-Za-z_][A-Za-z0-9_]*\b',
    'ANGKA': r'\b\d+\b',
    'OPERATOR': r'[+\-*/=:]',
    'BANDING': r'==',
    'SPASI': r'\s+',
    'TIDAK_DIKENAL': r'.'
}

def identifikasi_token(kata):
    for jenis_token, regex in JENIS_TOKEN.items():
        if re.fullmatch(regex, kata):
            return jenis_token, kata
    return 'TIDAK_DIKENAL', kata

def lexer(masukkan):
    token = []
    kata_sekarang = ''
    for karakter in masukkan:
        if karakter.isspace() or re.match(r'[+\-*/=:]', karakter):
            if kata_sekarang:
                jenis_token, nilai_token = identifikasi_token(kata_sekarang)
                token.append((jenis_token, nilai_token))
            kata_sekarang = karakter
    if kata_sekarang:
        jenis_token, nilai_token = identifikasi_token(kata_sekarang)
        token.append((jenis_token, nilai_token))
```

```

        kata_sekarang = ''
        if karakter.isspace():
            continue
        jenis_token, nilai_token = identifikasi_token(karakter)
        token.append((jenis_token, nilai_token))
    else:
        kata_sekarang += karakter

    if kata_sekarang:
        jenis_token, nilai_token = identifikasi_token(kata_sekarang)
        token.append((jenis_token, nilai_token))

    return token

# 51421476
masukkan = input("Berikan input: ")

token = lexer(masukkan)
for t in token:
    print (t)

```

1. JENIS_TOKEN = { ... }

Ini adalah kamus (*dictionary*) yang mendefinisikan berbagai pola yang digunakan untuk mengenali token dalam input. Pola-pola ini menggunakan *regular expressions*:

- 'KATA_KUNCI': Mencocokkan kata kunci tertentu seperti if, else, for, while, return.
- 'IDENTIFIKATOR': Mencocokkan identifikator atau nama variabel (huruf atau underscore diikuti huruf, angka, atau underscore).
- 'ANGKA': Mencocokkan angka (digit) yang terdiri dari satu atau lebih angka.

- 'OPERATOR': Mencocokkan operator seperti +, -, *, /, = atau :.
- 'BANDING': Mencocokkan operator perbandingan ==.
- 'SPASI': Mencocokkan spasi atau karakter kosong lainnya.
- 'TIDAK_DIKENAL': Mencocokkan karakter apapun yang tidak cocok dengan pola di atas.

2. def identifikasi_token(kata)

Fungsi ini bertugas untuk mengidentifikasi jenis token dari string kata yang diberikan. Fungsi ini membandingkan kata dengan pola-pola dari kamus JENIS_TOKEN menggunakan re.fullmatch. Jika cocok, maka dikembalikan pasangan (jenis_token, kata) yang sesuai. Jika tidak cocok, maka dikembalikan 'TIDAK_DIKENAL' sebagai jenis token.

3. def lexer(masukkan)

Fungsi ini adalah *lexer* yang berfungsi memecah input (string) menjadi token-token yang diidentifikasi jenisnya.

- token = []: Inisialisasi daftar kosong untuk menyimpan hasil token.
- kata_sekarang = "": Variabel ini menyimpan kata yang sedang dibangun selama proses iterasi karakter input.

4. for karakter in masukkan:

Fungsi ini melakukan iterasi karakter per karakter dari input pengguna.

- Jika karakter adalah spasi atau operator (+, -, *, /, =, :), maka program memproses token yang sudah terbentuk di variabel kata_sekarang dan menambahkan operator sebagai token baru.
- Jika karakter bukan spasi atau operator, ia ditambahkan ke variabel kata_sekarang untuk membentuk sebuah kata atau angka.

5. Setelah loop selesai

Jika masih ada kata yang tersisa di kata_sekarang, kata tersebut juga diidentifikasi dan dimasukkan sebagai token terakhir.

6. token = lexer(masukkan)

Masukan dari pengguna diproses oleh fungsi lexer dan hasilnya adalah daftar token.

7. for t in token: print(t)

Daftar token yang dihasilkan kemudian dicetak satu per satu.

Output nya didapatkan dikarenakan,

Ketika input yang diberikan adalah 5 1 4 2 + 1 4 7 6, lexer akan memproses setiap karakter satu per satu:

1. 5 diidentifikasi sebagai 'ANGKA' karena cocok dengan pola $\backslash b \backslash d + \backslash b$ (angka).
2. 1, 4, dan 2 juga diidentifikasi sebagai 'ANGKA' karena pola yang sama.
3. Spasi diabaikan.
4. Operator + dikenali sebagai 'OPERATOR' karena cocok dengan pola $r'[+ \backslash - * / = :]'$.
5. Spasi diabaikan lagi.
6. 1, 4, 7, dan 6 masing-masing diidentifikasi sebagai 'ANGKA'.