



begin forward
 $i = 1$
 $z = 0$
 else
 $z = 1$

بنام خدا

در تحلیل گری لغوی بسته را باید کاراکتر به کاراکتر بخوانیم و هر جا توکنی قابل استخراج دید، آن را استخراج کنیم و اعلام کنیم به مرحله بعد (ایسترا) ایسترا کنیم و در صورت نیاز به جدول محل اضافه می کنیم

look ahead

1.25 = 1.25 Do Si انتساب عددین

Do Si = 1, 5
 صله
 keyword
 تا اینجا باید look ahead داشته باشیم

Lexical Specification

شخص کردن الگوریتم توکن ← با استفاده از RE

$S \in L(R)$

$\Sigma = \{ \text{digit}^{(0..9)}, \text{letter } a..z, A..Z, (, (, -, >, <, \} \}$

مثال
 $R_1 \text{ Number} : \text{digit}^+$

$R_2 \text{ key} : 'if' + 'else' + \dots$

$R_3 \text{ id} : \text{letter} (\text{letter} + \text{digit})^*$

$R = \text{keyword} + \text{id} + \text{Number} + \dots$

مثال
 $G(R)$
 جدولی نیست

اگر ورودی $x_1 \dots x_n$

for $1 \leq i \leq n$ check $x_1 \dots x_i \in L(R)$?

if success, then we know that

$x_1 \dots x_i \in L(R_j)$ for some j

remove $x_1 \dots x_i$ from input and goto *

صورت دیگر ورودی برداریم؟

۱۲۴
 $x_1 \dots x_i \in L(R)$
 $x_1 \dots x_j \in L(R)$
 $i \neq j$

1250.12
 integer
 عدد صحیح

Maximal match : اصل



$x_1 \dots x_i \in L(R_j)$ $i \leq ?$
 $x_1 \dots x_i \in L(R_k)$ $i \leq ?$

راه حل اولویت دادن به کلمات

Lexical implementation

Regular Exp \rightarrow Specification

Finite auto \rightarrow implementation

FA : \sum الفبا , V مجموعه حالات , $s \in V$ حالت شروع , $F \subseteq V$ مجموعه حالات پذیرش , Rules (transitions) انتقال $s_1 \xrightarrow{a} s_2$

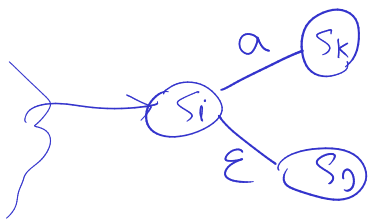
باید در هر لحظه از حالت شروع، آغاز می کنیم، اگر دردی نام نه در state خاصی باشیم \leftarrow accept



$1^* \emptyset$



DFA (epsilon move بدون , state برای ورودی در هر state)
 NFA (epsilon move داشته باشد , بیشتر از یک)



این DFA می تواند به NFA تبدیل شود

NFA می تواند به DFA تبدیل شود. NFA، بسته ای را می پذیرد که DFA می پذیرد. هر state نامی مشخص شود.

* DFA و NFA مجموعه یکسانی از زبان ها را تشخیص می دهند

بن سوال: تفاوت NFA و DFA برای ما در چیست؟

