



FLEX – Fast Lexical Analyzer

Kurzübersicht

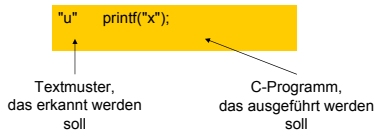
Karin Haenelt

Funktion

- Generator zur Erstellung von Programmen zur Erkennung von Textmustern
- Beschreibung von Textmustern durch reguläre Ausdrücke

Funktion

- Erkennung von Textmustern
- Ausführung von Aktionen



Format der Lex-Quelldatei

Beispiel

```

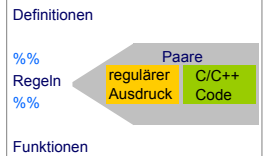
/*****
/* ein x für ein u vormachen */
/*****
%option noyywrap

%%
"u" printf("x");
%%

main () {
    yylex();
}

```

Struktur

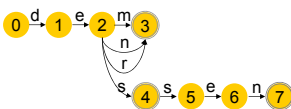


Überführung der regulären Ausdrücke in endliche Automaten

Regulärer Ausdruck

de([mnrsl]*ssen)

Endlicher Automat Graph

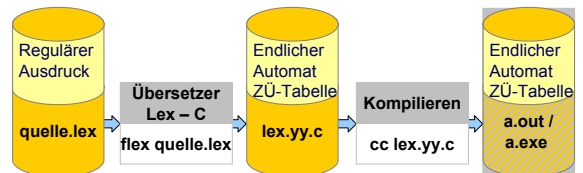


Endlicher Automat Zustandsübergangstabelle

	d	e	m	n	r	s
0	1					
1		2				
2			3	3	3	4
3			4			
4						5
5				6		
6					7	
7						

Erstellung eines Lex-Programms

Standardverarbeitung



Erstellung und Verwendung

- flex `Quelldatei.lex` C
 - (g)cc `lex.yy.c`
 - `a.out < Testdaten`
-
- flex `Test01.lex` %option outfile=„Test01.c“
 - (g)cc `-o test01 Test01.c`
 - `test01 < Testdaten`
-
- flex `Quelldatei.ll` C++
 - g++ `lex.yy.cc`
 - `a.out < Testdaten`

Konstrukt	Bedeutung
x	Das Zeichen "x"
"x"	Ein "x", auch wenn x ein Operator ist
\x	Ein "x", auch wenn x ein Operator ist
[xy]	Das Zeichen "x" oder "y"
[x-z]	Die Zeichen "x" bis "z"
[^x]	Beliebiges Zeichen außer "x"
.	Beliebiges Zeichen außer Zeilenendezeichen
^x	Ein x am Anfang einer Zeile
<y>x	Ein x, wenn Lex in Anfangsbedingung y ist
x\$	Ein x am Ende einer Zeile
x/y	Ein x, aber nur, wenn ein y folgt
x?	Ein optionales x
x*	0,1,2,... Wiederholungen von x
x+	1,2,3,... Wiederholungen von x
x{m,n}	m bis n Wiederholungen von x
x y	Ein x oder ein y
(x)	Ein x
{xx}	Die Ersetzung von xx aus dem Definitionsteil (Staubach,S.79)

Lex-Variablen

- `char yytext[]` enthält die erkannte Zeichenkette
- `int yyleng` enthält die Länge der Zeichenkette

Funktion	Bedeutung
<code>yylex()</code>	Funktion zur Steuerung der Erkennung regulärer Ausdrücke und der zugehörigen Aktionen; Definiert in <code>lex.yy.c</code> Benötigt die Funktion <code>main()</code> zum Starten aus der Shell-Ebene
<code>yymore()</code>	Das nächste erkannte Muster wird an <code>yytext</code> angehängt (default: überschreibt <code>yytext</code>)
<code>yyless(n)</code>	Verkürzt die in <code>yytext</code> stehende Zeichenkette auf n Zeichen
<code>unput(c)</code>	Schreibt c in den Eingabestrom zurück
<code>input()</code>	Liest das nächste Zeichen aus dem Eingabestrom
<code>yyterminate()</code>	Standardmäßig bei EOF aufgerufen, re-definierbar
<code>yywrap()</code>	EOF-Routine, kann im Skript definiert werden, andernfalls <code>%option noyywrap</code> erforderlich

Literatur

- Online: `man flex` oder `man lex`
- Bach, Fred; Dorman, Peter; Weng-Beckmann, Ulrike (1987) (Hrsg.): *UNIX™. Handbuch zur Programmentwicklung.* München, Wien: Carl Hanser Verlag.
- Staubach, Gottfried (1989): *Werkzeuge zur Textmusterverarbeitung. Awk, Lex und Yacc.* Berlin, Heidelberg, New York: Springer Verlag.