

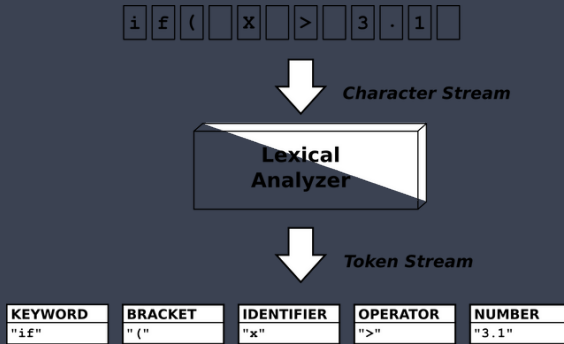
Analizador Léxico de C creado con Flex

Instituto Tecnológico de Costa Rica
Compiladores e Intérpretes
Semestre II 2020

by Ronald Herrera Gámez
on November 25, 2020

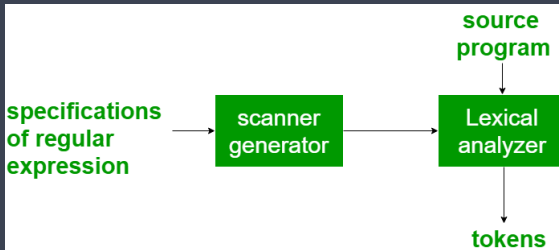
» Proceso de Scanning

Consiste en determinar las diferentes unidades elementales de un programa fuente, es decir, identifica los distintos lexemas de un lenguaje. Para este proceso, el scanner busca patrones dentro del fuente que cumplan con instrucciones, operadores, identificadores, constantes, entre otros, del lenguaje que se escanea.



» Flex (Fast Lexical Analyzer)

Flex es una herramienta para generar analizadores léxicos basados en la teoría de autómatas finitos. Para ello, se crea la descripción del escáner en forma de pares de expresiones regulares y código C, llamados reglas. Flex genera un archivo fuente en C llamado "lex.yy.c" que se puede compilar y vincular para producir un ejecutable. Este ejecutable analiza su entrada en busca de ocurrencias de texto que coinciden con las expresiones regulares para cada regla y siempre que encuentra una coincidencia, ejecuta el código C correspondiente.



» Código Después Del Preproceso

TOKENS:

KEYWORD

IDENTIFIER

LITERAL

OPERATOR

PUNCTUATOR

COMMENT

LEXICAL ERROR

PREPROCESSOR

```

#include <stdio.h> #include "octal_hex_2.c" /** * Probando numeros
reales, octales * y hexadecimales * @return */ int main ( ) { // Declara-
cion de un numero real float _n = -.5e+12 ; printf ( "%f \n" , _n ) ;
_n *= 2.e-1 ; /* multiplicado por */ printf ( "%f \n" , _n ) ; int hex-
adecimal = 0x31f ; /* representa 799 en decimal */ int hexadecimal_2 =
0X1a2 ; /* representa 418 en decimal */ printf ( "hexadecimal 1: %d\n"
, hexadecimal ) ; printf ( "hexadecimal 2: %d\n" , hexadecimal_2 ) ; hex-
adecimal_2 ++ ; hexadecimal += hexadecimal_2 ; printf ( "resultado:
%d\n" , hexadecimal ) ; // Salida 916 int octal_1 = 023 ; /* representa
19 en decimal */ int octal_2 = 0176 ; /* representa 126 en decimal */
printf ( "octal 1: %d\n" , octal_1 ) ; printf ( "octal 2: %d\n" , octal_2
) ; printf ( "resultado: %d\n" , ( octal_1 + octal_2 ) ) ; multiplicar ( ) ;
piramide ( 10 ) ; return 0 ; }

```

» Histograma Tokens Usados

