

## Proyecto 1

José Ceciliano  
Granados  
2016087245

Audra Rodríguez  
Mora  
2015101893

David Valverde  
Zuñiga  
200922986

# Proyecto 1

## Un generador de Scanners

José Ceciliano Granados  
2016087245

Audra Rodríguez Mora  
2015101893

David Valverde Zuñiga  
200922986

Instituto Tecnológico de Costa Rica  
Compiladores e Intérpretes  
I Semestre 2019

# Introducción

## Proyecto 1

José Ceciliano  
Granados  
2016087245

Audra Rodríguez  
Mora  
2015101893

David Valverde  
Zuñiga  
200922986

## Introducción

Flex es una herramienta de análisis lexico desarrollada para la generación de Scanners, programas que reconocen patrones léxicos en el texto.

Su nombre significa "fast lexical analyzer generator" y se encarga de leer las entradas recibidas para generar la descripción de un scanner en forma de pares de expresiones regulares y código C, llamadas "reglas".

# Scanning

## Proyecto 1

José Ceciliano  
Granados  
2016087245

Audra Rodríguez  
Mora  
2015101893

David Valverde  
Zuñiga  
200922986

## Scanning

Mediante proceso de Scanning se identifican los diferentes lexemas de un lenguaje. Flex genera como salida un archivo C que define una rutina en específico, que junto con una biblioteca genera un ejecutable con la capacidad de analizar su entrada para la aparición de expresiones regulares. Cada vez que encuentra una, ejecuta el código C correspondiente.

Todo esto se puede traducir a que Flex es capaz de crear un "Deterministic Finite Automaton" (DFA) que se utilizará para adquirir los diferentes lexemas que se pretenden escánear.

# Histograma I

## Proyecto 1

José Ceciliano  
Granados  
2016087245

Audra Rodríguez  
Mora  
2015101893

David Valverde  
Zuñiga  
200922986

## Histograma

A continuación se presenta un histograma que muestra la cantidad de cada tipo de *token* encontrado en el código fuente:

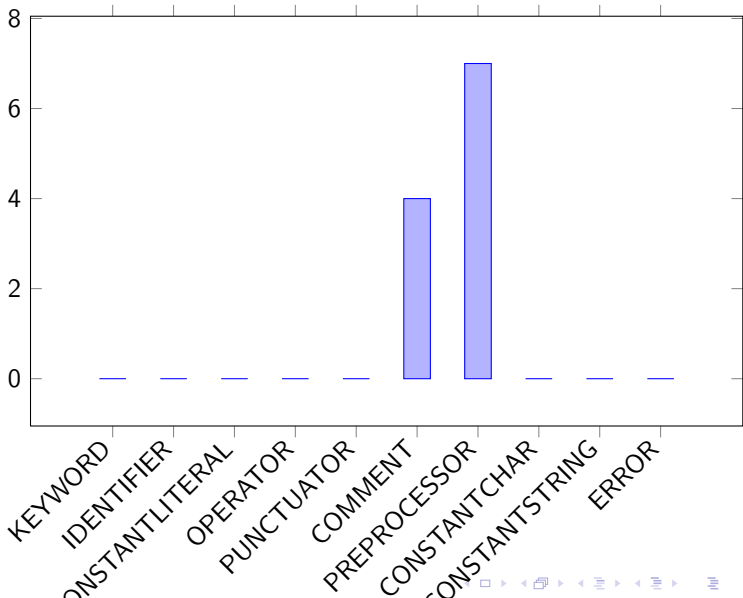
# Histograma

## Proyecto 1

José Ceciliano  
Granados  
2016087245

Audra Rodríguez  
Mora  
2015101893

David Valverde  
Zuñiga  
200922986



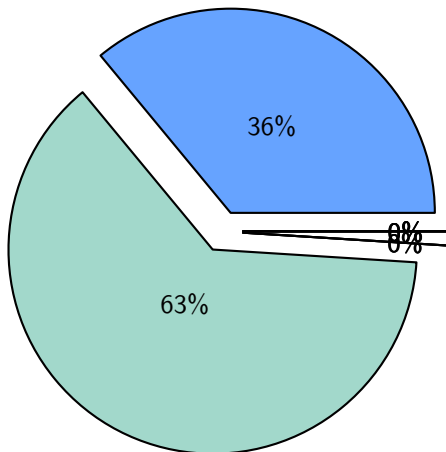
# Histograma

## Proyecto 1

José Ceciliano  
Granados  
2016087245

Audra Rodríguez  
Mora  
2015101893

David Valverde  
Zuñiga  
200922986



- KEYWORD-0
- IDENTIFIER-0
- CONSTANTLITERAL-0
- OPERATOR-0
- PUNCTUATOR-0
- COMMENT-4
- PREPROCESSOR-7
- CONSTANTCHAR-0
- CONSTANTSTRING-0
- ERROR-0

# Analisis Léxico I

## Proyecto 1

José Ceciliano  
Granados  
2016087245

Audra Rodríguez  
Mora  
2015101893

David Valverde  
Zuñiga  
200922986

## Codigo fuente

A continuación se presenta el código fuente con colores demostrando la división de *tokens*.

# Resaltado de sintaxis I

## Proyecto 1

José Ceciliano  
Granados  
2016087245

Audra Rodríguez  
Mora  
2015101893

David Valverde  
Zuñiga  
200922986

```
# 1 "tests/1.c"
# 1 "<built-in>"
# 1 "<command-line>"
# 31 "<command-line>"
# 1 "/usr/include/stdc-predef.h" 1 3 4

# 1 "/usr/include/stdc-predef.h" 3 4
```

```
/* Copyright (C) 1991-2018 Free Software Foundation, Inc.
   This file is part of the GNU C Library.
```

The GNU C Library is free software; you can redistribute and modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later



# Resultado de sintaxis II

## Proyecto 1

José Ceciliano  
Granados  
2016087245

Audra Rodríguez  
Mora  
2015101893

David Valverde  
Zuñiga  
200922986

The GNU C Library is distributed in the hope that it will be useful to you, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with the GNU C Library; if not, see <http://www.gnu.org/licenses/>. \*/

# Resaltado de sintaxis III

## Proyecto 1

José Ceciliano  
Granados  
2016087245

Audra Rodríguez  
Mora  
2015101893

David Valverde  
Zuñiga  
200922986

```
/* This header is separate from features.h so that the compiler  
   include it implicitly at the start of every compilation  
   not itself include <features.h> or any other header that  
   <features.h> because the implicit include comes before  
   test macros that may be defined in a source file before  
   explicitly includes a system header.  GCC knows the name  
   header in order to preinclude it. */
```

```
/* glibc's intent is to support the IEC 559 math functions  
   and complex.  If the GCC (4.9 and later) predefined macros  
   specifying compiler intent are available, use them to decide  
   whether the overall intent is to support these features  
   presume an older compiler has intent to support these features  
   define these macros by default. */
```

# Resultado de sintaxis IV

## Proyecto 1

José Ceciliano  
Granados  
2016087245

Audra Rodríguez  
Mora  
2015101893

David Valverde  
Zuñiga  
200922986

```
# 52 "/usr/include/stdc-predef.h" 3 4
```

```
/* wchar_t uses Unicode 10.0.0. Version 10.0 of the Unicode  
synchronized with ISO/IEC 10646:2017, fifth edition, plus  
the following additions from Amendment 1 to the fifth edition:  
- 56 emoji characters  
- 285 hentaigana  
- 3 additional Zanabazar Square characters */
```

```
/* We do not support C11 <threads.h>. */
```

# Resultado de sintaxis V

## Proyecto 1

José Ceciliano  
Granados  
2016087245

Audra Rodríguez  
Mora  
2015101893

David Valverde  
Zuñiga  
200922986

```
# 32 "<command-line>" 2  
# 1 "tests/1.c"
```

```
# 1 "tests/1.c"  
main() {  
    int a;  
}
```