|  |  |  |
| --- | --- | --- |
| CASTRO CRUCES JORGE E. | **INSTITUTO POLITÉCNICO NACIONAL** | **ESCUELA SUPERIOR DE CÓMPUTO** |
| **SEGUNDO PARCIAL** | **PRACTICA 3 – INTRODUCCIÓN A FLEX** | **30/04/2021** |
| 3CV18 | **COMPILADORES** | **Hernández Olvera Luis E** |

**SECCIÓN TEÓRICA O INTRODUCCIÓN**

**Introducción a Flex:**

• Flex es una herramienta que permite generar escáneres: (programas que reconocen patrones léxicos en un texto).

• A partir de un conjunto de expresiones regulares, Flex busca concordancias en un fichero de entrada y ejecuta acciones asociadas a estas expresiones.

Flex ( generador de analizador léxico rápido ) es una alternativa de software libre y de código abierto a lex . [2] Es un programa informático que genera analizadores léxicos (también conocidos como "scanners" o "lexers"). [3] [4] Se utiliza con frecuencia como la implementación de lex junto con el generador de analizador sintáctico Berkeley Yacc en sistemas operativos derivados de BSD (ya que ambos lexy yaccson parte de POSIX ), [5] [6] [7] o junto con GNU bison (una versión de yacc) en puertos \* BSD [8] y en distribuciones de Linux. A diferencia de Bison, flex no es parte del Proyecto GNU y no se publica bajo la Licencia Pública General GNU , [9] aunque la Free Software Foundation produjo y publicó un manual para Flex. [10]

**Problemáticas:**

* Complejidad del tiempo

Un analizador léxico Flex generalmente tiene complejidad de tiempo O(n) en la longitud de la entrada. Es decir, realiza un número constante de operaciones para cada símbolo de entrada. Esta constante es bastante baja: GCC genera 12 instrucciones para el ciclo de coincidencia de DFA. Tenga en cuenta que la constante es independiente de la longitud del token, la longitud de la expresión regular y el tamaño del DFA.

Sin embargo, el uso de la macro REJECT en un escáner con el potencial de hacer coincidir tokens extremadamente largos puede hacer que Flex genere un escáner con un rendimiento no lineal. Esta característica es opcional. En este caso, el programador le ha dicho explícitamente a Flex que "regrese y vuelva a intentarlo" después de que ya haya coincidido con alguna entrada. Esto hará que DFA retroceda para buscar otros estados de aceptación. La función RECHAZAR no está habilitada de forma predeterminada y, debido a sus implicaciones en el rendimiento, se desaconseja su uso en el manual de Flex.

* Reentrada

De forma predeterminada, el escáner generado por Flex no es reentrante . Esto puede causar serios problemas para los programas que utilizan el escáner generado desde diferentes subprocesos. Para superar este problema, Flex ofrece opciones para lograr la reentrada. Puede encontrar una descripción detallada de estas opciones en el manual de Flex.

* Uso en entornos que no son Unix

Normalmente, el analizador generado contiene referencias al archivo de encabezado unistd.h que es específico de Unix . Para evitar generar código que incluya unistd.h , se debe usar la opción% option nounistd . Otro problema es la llamada a isatty (una función de la biblioteca de Unix), que se puede encontrar en el código generado. La opción% nunca interactiva obliga a flex a generar código que no usa isatty .

* Uso de flex desde otros idiomas

Flex solo puede generar código para C y C ++ . Para utilizar el código del escáner generado por flex desde otros idiomas , se puede utilizar una herramienta de vinculación de idiomas como SWIG .

**Flex++**

Flex ++ es un escáner léxico similar para C ++ que se incluye como parte del paquete flex. El código generado no depende de ningún tiempo de ejecución o biblioteca externa , excepto de un asignador de memoria ( malloc o una alternativa proporcionada por el usuario) a menos que la entrada también dependa de él. Esto puede resultar útil en situaciones integradas y similares en las que el sistema operativo tradicional o las funciones de tiempo de ejecución de C pueden no estar disponibles.

El escáner C ++ generado por flex ++ incluye el archivo de encabezado FlexLexer.h, que define las interfaces de las dos clases generadas por C ++.

**PROBLEMAS RELACIONADOS AL TEMA A LOS CUALES TE ENFRENTASTE AL PROGRAMAR LA PRÁCTICA**

Para esta práctica, el principal problema al que me enfrenté fue aprender y comprende run nuevo lenguaje; Porque Flex es prácticamente un lenguaje de programación de escáneres de patrones léxicos.

Por supuesto, me tuve que apoyar en libros y sobre todo en videotutoriales de YouTube, para poder comprender de mejor forma el lenguaje.

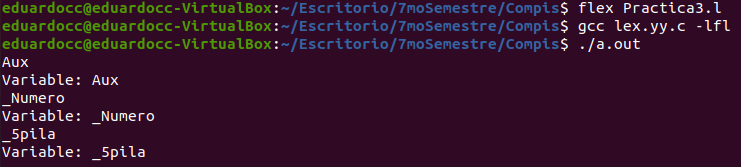
Algo que me ayudo mucho fue guiarme de ejemplos que encontré en la web, ya que eran ejercicios simples y con gran valor educativo.

Sin duda, otro gran problema al que me enfrenté fue la instalación de la librería Flex en la terminal de Linux ,ya que tuve algunas complicaciones a la hora de instalar el comando. Mismas que pude superar con ayuda de foros dedicados a responder y apoyar a estudiantes o entusiastas de la programación en Flex.

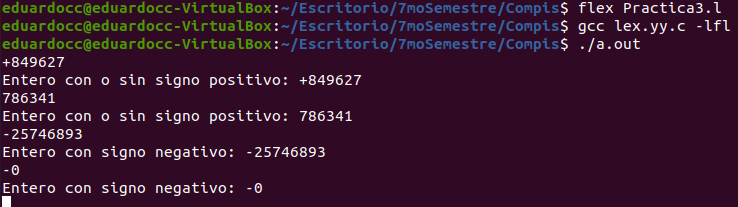
**CAPTURAS DE PANTALLA**

El objetivo de la práctica es diseñar las expresiones regulares en Flex que reconozcan:

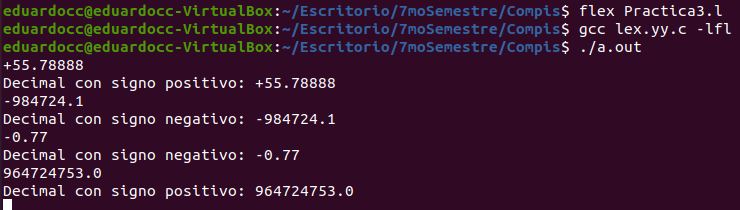
* Nombres de variables.



* Números enteros con y sin signo.



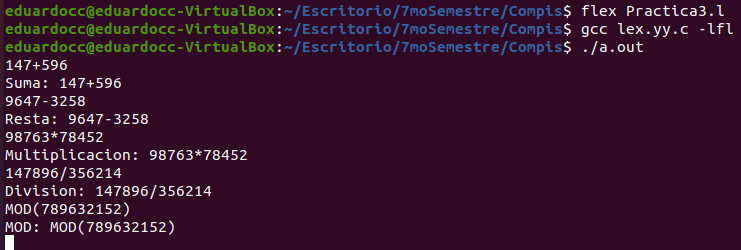
* Números en forma decimal con y sin signo.



* Potencia (con signo o sin él) de números reales (con signo o sin él) .



* Operaciones matemáticas: +, -, \*, /, MOD()



**CONCLUSIÓN**

**¿Qué aprendí de esta práctica?**

Claramente, aprendí muchas cosas, entre ellas está:

* Implementar archivos .l
* Programar escáneres de patrones Léxicos en Flex
* Instalar la herramienta Flex en la terminal de Linux
* Aprender un nuevo lenguaje de programación

**¿En qué me ayudó esta práctica?**

Claramente, me ayudo a comprender de mejor forma la utilización de la herramienta Flex, y el gran poder que posee.

También, expandió mi conocimiento sobre la materia de Compiladores y el abanico de posibilidades que esta herramienta nos otorga.

**REFERENCIAS**

[2]Levine, John R .; Mason, Tony; Brown, Doug (1992). lex y yacc (2ª ed.). O'Reilly . pag. 279. ISBN 1-56592-000-7. Una versión de lex disponible gratuitamente es flex .

[3]Levine, John R .; Mason, Tony; Brown, Doug (1992). lex y yacc (2ª ed.). O'Reilly . págs. 1-2. ISBN 1-56592-000-7.

[4]Levine, John (agosto de 2009). flex y bison . O'Reilly Media. pag. 304. ISBN 978-0-596-15597-1.

[5]OpenBSD (11 de diciembre de 2015). "src / usr.bin / lex /" . Referencia cruzada BSD . Consultado el 26 de diciembre de 2015 . Esto es flex, el generador de analizador léxico rápido.

[6]" " . \* Páginas de manual de BSD .flex(1)

[7]" " . \* Páginas de manual de BSD .yacc(1)

[8]"bison-3.0.4 - generador de analizador de GNU" . Puertos OpenBSD . 2015-11-15 . Consultado el 26 de diciembre de 2015 .

[9] ¿Flex GNU es o no? Archivado el 3 de marzo de 2016 en Wayback Machine , preguntas frecuentes sobre flex

[10]"Flex - un generador de escáner - Tabla de contenido - Proyecto GNU - Free Software Foundation (FSF)" . ftp.gnu.org . Consultado el 5 de diciembre de 2019 .