

Comentario

Abecedario (Sigma)

Sigma = {/, cualquier otro carácter}

Palabras (W)

W =

Regex = `/[\V|\w|\s]*/` (incluidos caracteres especiales)

Lenguaje (L)

L = {// comentario, }

Regex = `/[V]{2}.*$/`

- \n sale con qf

Diagrama de Moore

Modelo Matemático

Tabla de Transiciones

Comentario multilínea

Abecedario (Sigma)

Sigma = {/, *, cualquier otro carácter}

Palabras (W)

W =

Regex = `/[\\|*|\\w|\\s]*/` (incluidos caracteres especiales)

Lenguaje (L)

L = {/* comentario multilínea */, }

Regex = `\\^*([\\s\\S]*?)*\\^/`

- `\\s` permite incluir los saltos de línea en los comentarios
- `?` delimita el comentario por `/*` y `*/` e impide que el código que está dentro de dos comentarios multilínea sea interpretado como comentario
- `*/` sale con q_f

Diagrama de Moore

Modelo Matemático

Tabla de Transiciones

Variables

Abecedario (Sigma)

$\Sigma = \{w, _, \text{cualquier otro símbolo}\}$

Palabras (W)

$W =$

$\text{Regex} = /[\w_]* /$ (incluidos caracteres especiales)

Lenguaje (L)

$L = \{s, n, \text{string}, \text{num}, \text{Num1}, \dots\}$

$\text{Regex} = /[A-Za-z_][\w_]* /$

- los nombres de las variables deben de empezar con una letra mayúsculas o minúsculas o con guión bajo
- después de la primera, puede seguir cualquier símbolo

Diagrama de Moore

Modelo Matemático

Tabla de Transiciones

cin

Abecedario (Sigma)

Sigma = {c, i, n, >, cualquier otro carácter}

Palabras (W)

W =

Regex = `/[\w|\s]*/` (incluidos caracteres especiales)

Lenguaje (L)

L = {`// comentario,`}

Regex = `/cin[\t|]*>[\t|]*[A-Za-z_][\w_]*;/`

- `[\t|]*` puede aceptar cualquier número de espacios entre los componente sintácticos
- está incluido el “token” de variables

Diagrama de Moore

Modelo Matemático

Tabla de Transiciones

cout

Abecedario (Sigma)

Sigma = {c, o, u, t, " , < , ; , cualquier otro carácter}

Palabras (W)

W =

Regex = `/[\w|\s|<|\"|;]*` / (incluidos caracteres especiales)

Lenguaje (L)

L = {cin << "Name : " << name; , ...}

Regex = `/cout[\t|]*(<<[\t|]*([A-Za-z_][\w_]*|\".*\"))[\t|]*);/`

- `\".*\"` permite también recibir strings entre comillas

Diagrama de Moore

Modelo Matemático

Tabla de Transiciones

Especificaciones de diseño

Un programa principal será el que lea del archivo de texto línea por línea. Cada línea entrará de manera secuencial a un programa que será el linter; el analizador sintáctico. Este se encargará de detectar alguna de las siguientes cadenas:

- `//` manda al autómata de `comentario`
- `/*` manda al autómata de `comentario multilínea`
- `cin` manda al autómata del mismo nombre
- `cout` manda al autómata del mismo nombre

Una vez el control del programa salte a alguna de estas secciones, se evaluará la línea (o las líneas) correspondientes hasta salir con `qf` o `qe`

- si sale con `qf` el analizador sintáctico recibe otra línea y se repite el ciclo de detección
- si sale con `qe` el analizador sintáctico termina con el estado de error, ya no recibe más líneas