Análisis lexicográfico

Ing. Karina Cabrera Chagoyan

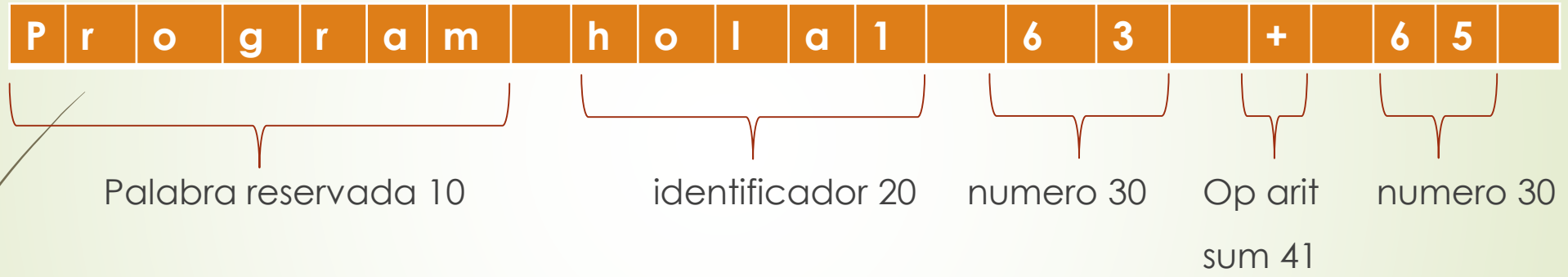
Funciones del analizador lexicográfico

- Su función principal es agrupar los caracteres que va leyendo uno a uno el programa fuente y formar los tokens
- Otras funciones
  - Gestionar el archivo que contiene el código fuente (abrir, leer, cerrar)
  - Eliminar comentarios , tabuladores, espacios en blanco, saltos de línea
  - Relacionar los errores con las líneas del programa
  - Introducir identificadores en tabla de símbolos (puede realizarla el análisis sintáctico)

Ejemplo

Program hola1 63 + 65

- Entrada 22 caracteres



- Salida 5 tokens

Definiciones básicas

- Token
- Lexema
- Patrón

Token

- Es una agrupación de caracteres reconocidos por el analizador lexicográfico que constituyen los símbolos con los que se forman las sentencias del lenguaje y también se les conoce como **componentes léxicos**
- Son los **SIMBOLOS TERMINALES** en una **gramática**
  - **Palabras reservadas**
  - **Identificadores**
  - **Operadores y constantes**
  - **Símbolos de puntuación y especiales**

Lexema

- Es la secuencia de caracteres, ya agrupados , que coinciden con un determinado token, como por ejemplo

  El nombre de un identificador o el valor de un numero

  Un token puede tener uno o infinitos lexemas . Por ejemplo

  Las palabras reservdas tienen un solo lexema , mientras que los identificadores o los numeros tienen infinitos

Patrón

- Es la forma de describir los tipos de lexemas . Se puede definir utilizando
  - Expresiones regulares
  - Autómatas finitos deterministas
  - Descripción informal
- Ejemplo

Token (componente léxico)	Lexema	Patron
If	If	If
Operador aritmético de multiplicación	*	*
Identificador	X, numero, variable	[a-zA-Z]+

Como funciona el analizador léxico

- El analisis léxico funciona bajo demanda del analizador sintáctico cuando le pide el siguiente token .
- A partir de este archive que contiene el codigo Fuente se lee caracter por caracter, que son almacenados en un buffer de entrada.

Diseño de un analizador léxico

- Para poder construir un analizador léxico primero se debe diseñar , se puede usar una tabla o un diagrama de transición que representa los estados por los que va pasando el autómata para reconocer un token

Reconocimiento de tokens

- Una vez que se diseña un analizador a través del diagrama o tabla de transición se reconocen los tokens
- Por ejemplo
- Suponga un identificador conformado por al menos una letra mayúscula o minúscula seguida de manera opcional por mas letras o numeros (xy, num1, etc)
- Utilizando un diagrama de transición y expresión regular para representar una letra : [a-z][A-Z] y le llamamos letra , para representar un numero cualquiera [0-9] y le llamamos numero y por ultimo definimos [otro] como cualquier otro símbolo

Implementación de una analizador léxico

- Hay varias formas de implementar un analizador léxico

1. Utilizando un generador de analizadores léxicos

- Son herramientas que a partir de las expresiones regulares generan un programa que permite reconocer los tokens o componentes léxicos
- Estos programas suelen estar escritos en C , donde una de las herramientas es FLEX o pueden estar escritos en JAVA , donde las herramientas pueden ser JFLEX o J LEX
- **Ventajas** : Comodidad y rapidez de desarrollo
- **Desventajas** : Estos programas son ineficientes y dificultad de mantenimiento del código generado

2. Utilizando un lenguaje de alto nivel

A partir del diagrama de transiciones y del pseudocodigo correspondiente se programa el analizador léxico

- **Ventajas**
  - Eficiente y compacto (lo que facilita el mantenimiento)
- **Desventajas**
  - Se debe realizar de forma manual

3. Utilizando un lenguaje de bajo nivel (ensamblador)

- **Ventajas**
  - Mas eficiente y compacto
- **Inconveniente**
  - Mas difícil de desarrollar

Errores léxicos

- Los errores léxicos son detectados , cuando durante el proceso de reconocimiento de caracteres , los símbolos que tenemos en la entrada no concuerdan con ningún patrón.
- **Algunos errores que pueden ser detectados son** :
  - Nombres incorrectos de identificadores
  - Números incorrectos
  - Palabras reservadas escritas incorrectamente
  - Caracteres que no pertenecen al alfabeto del lenguaje

Estrategias de recuperación de errores

- La estrategia mas sencilla es la que se conoce como recuperación **modo pánico**
- Intentar transformaciones para reparar la entrada es muy costoso. Otra estrategia aplicada es la siguiente :
  - Cuando se detecta el error y ya se ha pasado por un estado de aceptación , se ejecuta la acción correspondiente al estado de aceptación por la que se paso y el resto de caracteres se devuelve a la entrada y se posiciona en el estado inicial , para iniciar el siguiente reconocimiento de token
  - Si no se ha pasado por ningún estado de aceptación , se elimina el carácter que no concuerda y se acepta el siguiente

Modo pánico

- Consiste en borrar de forma sucesiva caracteres hasta que el analizador léxico es capaz de encontrar un token bien formado

  ?cual seria el proceso para tratar el error?

  - Anotar el error y el estado
  - Recuperarse . Se tienen varias alternativas : borrar, ignorar, reemplazar o insertar
  - Seguir