UNIVERSIDAD DE MÁLAGA

Procesadores de Lenguajes

PARCIAL 3 (2022)

Apartado 1. Implementar mediante JFlex y CUP un analizador sintáctico que permita evaluar expresiones de cadenas de caracteres. A continuación se muestra un ejemplo de programa en este lenguaje:

```
a = "hola";
b = "Adios";
c = a + b;
print( c );
a = c.substr( 1, 4 );
print( a );
c = c + "Carlos";
b = c.substr(4,8) + "s";
print(b);
print( a + c );
```

Este programa mostraría por pantalla:

```
holaAdios
olaA
AdiosCars
olaAholaAdiosCarlos
```

El lenguaje contendrá las siguientes operaciones:

- El lenguaje permite asignar cadenas a variables mediante el operador =. Las variables no es necesario definirlas anteriormente. El único tipo de variable que admite el lenguaje son las cadenas de caracteres.
- Los nombres de variables tienen que comenzar por una letra (mayúscula o minúscula) y seguir con cualquier combinación de letras, dígitos y carácter subrayado.
- El lenguaje cuenta con la función **print** para imprimir una cadena en pantalla. Esta función recibe un único parámetro que es una cadena de caracteres y la muestra en pantalla seguida de un retorno de carro/avance de línea.
- Las cadenas de caracteres pueden ser:
  - Una cadena constante entre comillas dobles.
  - Una variable.
  - El resultado de aplicar un operador a una cadena de caracteres.
- Los operadores válidos para cadenas de caracteres son:
  - La concatenación de cadenas con el operador +.
  - Obtener una subcadena de otra con el operador **.substr(p, l)**. Este operador se coloca después de una cadena y devuelve la subcadena desde la posición p con longitud l. Donde p y l son dos números.
- Un número puede ser:
  - Un valor literal constante.
- Las sentencias de este lenguaje terminan en un punto y coma.

Apartado 2. Añadir al lenguaje del apartado 1 las siguientes funciones:

- Obtener la longitud de una cadena con el operador **.size()**. Este operador se coloca después de una cadena y devuelve un número correspondiente a su longitud. Por ejemplo: c.size()
- La posibilidad de agrupar operaciones con cadenas mediante paréntesis. Por ejemplo: (a + b). substr(2, 3) o (a + "hola").size()
- Lo números de la operación **substr** pueden ser resultado de una expresión numérica. De esta forma, un número en este lenguaje pasa a ser una de las siguientes posibilidades:
  - Un valor constante.
  - El valor devuelto por el operador .size().
  - Una operación de suma o resta sobre dos números.

De esta forma, un ejemplo del lenguaje ampliado podría ser el siguiente:

```
a = "Hola";
b = "Adios";
c = (a + b).substr( 1, 3 );
c = a + c;
print( c );
d = c.substr( c.size() - 3, 3 );
print( d );
d = (d + a).substr( 0, (d + a).size() - 3 );
print( d );
```

Que mostraría por pantalla:

```
Holaola
ola
olaH
```