

Memoria de prácticas

Intérprete en Bison para lenguaje en pseudocódigo

Fernando Sánchez Delgado

Asignatura de Procesadores de Lenguajes
Grado en Ingeniería Informática
Especialidad en computación

Escuela Politécnica Superior
Universidad de Córdoba
Curso 2016-2017
20 Mayo 2017, Córdoba

Contents

1	Introducción	2
2	Language pseudocódigo	2
2.1	Componentes léxicos	2
2.1.1	Palabras reservadas	2
2.1.2	Identificador	2
2.1.3	Número	2
2.1.4	Cadena	3
2.2	Sentencias	3
3	Tabla de símbolos	3

1 Introducción

Este trabajo consiste en la realización de un intérprete para un lenguaje de pseudocódigo cuya especificación nos ha sido dada en el enunciado de la práctica. Asimismo se han añadido ciertas mejoras que se han visto oportunas para obtener tanto un lenguaje como un intérprete más robustos.

El intérprete admite dos modos de ejecución distintos, uno es el modo interactivo, el cual permite introducir código que será evaluado de inmediato. El otro modo consiste en proporcionar al intérprete un archivo del cual obtener el código a ejecutar.

En cuanto a la estructura de este documento, primero procederemos a hablar sobre el lenguaje, para después ir profundizando en los detalles de la implementación. Por último, se mostrarán algunos ejemplos y se expondrán las conclusiones obtenidas tras la realización de este trabajo.

2 Language pseudocódigo

2.1 Componentes léxicos

2.1.1 Palabras reservadas

Las palabras reservadas del lenguaje no podrán ser utilizadas como identificadores, se han reservado las siguientes palabras:

`_mod, _div, _o, _y, _no, leer, leer_cadena, escribir, escribir_cadena
si, entonces, si_no, fin_si, mientras, hacer, fin_mientras, repetir, hasta
para, desde, hasta, paso, fin_para, _borrar, _lugar`

2.1.2 Identificador

Los identificadores pueden estar compuestos de letras, dígitos o guiones bajos. Sin embargo, se debe cumplir que: empiece por una letra, no acabe en guión bajo y no haya dos guiones bajos seguidos.

Ejemplo de identificador válido:

`iden, iden_1a, ident_a_1_3`

Ejemplo de identificadores no válidos:

`_iden, iden_1_, iden__3`

2.1.3 Número

Los números pueden ser de tipo entero, real o con notación científica.

Ejemplo números válidos:

38 5.289 3.1e10

Ejemplo números no válidos:

3.000.000 3,895 0x398

2.1.4 Cadena

Las cadenas se delimitan por comillas simples y pueden contener cualquier tipo de carácter a excepción de un salto de línea. Para permitir caracteres especiales se han incluido los siguientes código de escape:

<code>\t</code>	Tabulador
<code>\n</code>	Salto de línea
<code>\'</code>	Comilla simple
<code>\\</code>	Barra invertida

Además de las cadenas simples, también se ha añadido un nuevo tipo de cadena, la cadena multilínea. Esta cadena permite la inclusión de saltos de línea. Su sintaxis requiere de tres comillas simples delimitando cada extremo de la cadena.

Ejemplo de cadenas válidas:

```
'Cadena simple'  
'\tCadena con caracteres escapados\n'  
,,  
Cadena  
multilínea'''
```

2.2 Sentencias

3 Tabla de símbolos