

Compilador Kaskell

Álvaro García Tenorio *

Miguel Pascual Domínguez **

19 de junio de 2018

Índice general

Índice general	1
1 Introducción	2
2 Estructura del compilador	3
2.1. Léxico	3
2.2. Sintáctico	3
2.3. Identificadores	3
2.4. Tipos	3
2.5. Generación Código de la Máquina-P	3
2.6. Errores	3
2.7. Ejemplo de Código	3

* alvgar14@ucm.es

** miguepas@ucm.es

Capítulo 1

Introducción

Comencemos primero explicando en que consiste nuestro lenguaje.

Un programa en código kaskell, consiste en 3 partes diferenciadas. Una primera parte que consiste en la declaración de registros. Nuestro lenguaje, permite la declaración de varios registros seguidos, que irán declarados uno detrás de otro, pueden estar separados por saltos de línea, espacios o sin separación. Además se permite la declaración de registros anidados, es decir, que uno de los campos de un registro, sea un registro previamente declarado.

La segunda parte consiste en los bloques. En nuestro lenguaje, los bloques están formados por la composición secuencial de instrucciones. Se permiten los bloques anidados, y las instrucciones pueden ser declaraciones de un array, un kinkeger (int) o un kool (bool), con o sin inicialización (asignación); asignaciones, khile (while), kor (for), if (if), kelse (else) o la llamada a una función.

Y por último, la tercera parte que consiste en la declaración de funciones. Las funciones pueden o no devolver un valor de tipo simple, es decir, un kinkeger o un kool, consideraremos que las funciones está formadas por dos partes, la parte de la cabeza que contiene los tipos de los argumentos, si los tiene, el identificador de la función y el tipo de retorno, si lo tiene; y la cola, que será los argumentos de tenerlos, el bloque de código y el valor de retorno, si lo tiene.

Capítulo 2

Estructura del compilador

Comentaremos ahora, la estructura del compilador, al igual del código del mismo. Para ello se comentará brevemente cada apartado del compilador.

2.1. Léxico

Comencemos con el léxico, en el léxico procesamos los identificadores de cualquier elemento del código, al igual que los símbolos especiales como "τ", y también las palabras reservadas.

Las palabras reservadas son: kif, kelse, khile, kor, kinkeger, kool, X (producto cartesiano), korr (or lógico), kand (and lógico), knot (not lógico), kod (operación modulo), kreturn (palabra reservada para el retorno de funciones), krue (true), kalse (false), y struk (registros).

Para poder analizar las palabras reservadas, nos ayudamos de un HashMap que tiene las palabras reservadas en el mismo, y al encontrar un identificador, comprobamos si es palabra reservada o no, viendo si está en el HashMap.

2.2. Sintáctico

La declaración se realiza de la siguiente manera: `identificador := struk declaraciones`

2.3. Identificadores

2.4. Tipos

2.5. Generación Código de la Máquina-P

2.6. Errores

2.7. Ejemplo de Código