

Universidad Rey Juan Carlos

Procesadores de lenguajes

Práctica Obligatoria. Parte 1: Análisis léxico y sintáctico

Autores: Sergio Paniego Blanco

Héctor Domínguez Diego

Descripción de la práctica

En la práctica hemos realizado la especificación léxica, en la que hemos detectado los identificadores, las constantes numéricas, que pueden ser del tipo real o decimal y con base decimal, octal o hexadecimal, las constantes literales y los comentarios.

Para la sintaxis, hemos definido las declaraciones de funciones y procedimientos, declaraciones de variables, asignaciones y llamadas a las funciones y procedimientos.

Además de esto, hemos definido las sentencias de control de flujo if, for y while, y los tipos de datos para distintas estructuras y matrices.

Para el control de errores, hemos detectado si no se ha escrito un principio de comentario y se ha puesto un final de comentario, si se ha declarado mal un número hexadecimal, tanto dejándolo a medias, como si se ha puesto más de un signo o más de un punto y si se ha abierto de forma incorrecta un comentario o si se ha cerrado de forma incorrecta.

Casos de uso

Caso correcto 1

Declaramos un registro, una función con instrucciones simples y un procedimiento con instrucciones de flujo y en la función main hacemos uso de estos. Estas declaraciones se pueden observar en el fichero prueba1.c.

La salida tras compilar este fichero se encuentra en el fichero resultado.txt, en él mostramos la ristra de tokens que se han detectado en el fichero prueba1.c.

Caso correcto 2

Declaramos una función en la que realizamos operaciones con las distintas bases y tipos de números y la función main donde hacemos uso de la función anterior.

La salida tras compilar este fichero se encuentra en el fichero resultado2.txt, en él mostramos la ristra de tokens que se han detectado en el fichero prueba2.c.

Caso correcto 3

Declaramos un registro, una función con instrucciones simples y un procedimiento con instrucciones de flujo y finalmente hacemos las llamadas a estos métodos en la función main.

La salida tras compilar este fichero se encuentra en el fichero resultado3.txt, en él mostramos la ristra de tokens que se han detectado en el fichero prueba3.c.

Caso correcto 4

Declaramos un registro, una función con instrucciones simples y un procedimiento con instrucciones de flujo que hace uso de las estructuras que hemos declarado y un procedimiento con instrucciones simples y finalmente hacemos las llamadas a estos métodos en la función main.

La salida tras compilar este fichero se encuentra en el fichero resultado4.txt, en él mostramos la ristra de tokens que se han detectado en el fichero prueba4.c.

Caso Error 1

Declaramos un registro, una función con instrucciones simples y un procedimiento con instrucciones de flujo que hace uso de las estructuras que hemos declarado y un procedimiento con instrucciones simples y finalmente hacemos las llamadas a estos métodos en la función main.

Caso error 2:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  | | --- | | struct registro { | |  | int campo\_a; | |  | float campo\_b; | |  | }; | |  |  | |  | void intercambio (int v1, int v2){ | |  | int auxiliar; | |  |  | |  | auxiliar = v1; | |  | v1 = v2; | |  | v2 = auxiliar; | |  | } | |  |  | |  | float formula (float p1, int p2){ | |  | float auxiliar; | |  |  | |  | intercambio( p1 \* 10, p2); | |  | while(p1 == 10.5 \* p2){ | |  | if( auxiliar < 10.5 and auxiliar >= 8.6 or not auxiliar == 9) then { | |  | auxiliar = auxiliar + -3 + p2 % 045 \* 0-3 / 0+67; | |  | auxiliar = auxiliar - (+98 - p1 \* 0x34 % 0xFC4D); | |  | } else { | |  | auxiliar = -3.98 + p2 % 045.27 \* 0-3.11 / 0+67.34; | |  | } | |  | auxiliar = auxiliar - (+98.5 - p1 \* 0x+ABD.EE % 0xFC4D.F4); | |  | } | |  | return auxiliar \* 0x-FF; | |  | } | |  |  | |  | void Main (int args){ | |  | int entero1, entero2; | |  | float real1[10][20], real2; | |  | //Aqui esta el error al cerrar el comentario mal | |  | \*/ | |  | for( entero1 = 1 ; entero1 < 21 ; entero1 = entero1 + 1){ | |  | entero2 = 1; | |  | do { | |  | real1[1][10] = 1.0; | |  | entero2 = entero2 + 1; | |  | } until (entero2 == 11) | |  | } | |  | printf('Esto es el programa \'principal\'...'); | |  | { | |  | intercambio(entero2, entero1); | |  | real1[1][1] = formula(real2,entero1 + registro.campo\_a); | |  | } | |  | registro.campo\_b = real1[3][1]; | |  | } | |

Caso error 3:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  | | --- | | struct registro { | |  | int campo\_a; | |  | float campo\_b; | |  | }; | |  |  | |  | void intercambio (int v1, int v2){ | |  | int auxiliar; | |  |  | |  | auxiliar = v1; | |  | v1 = v2; | |  | v2 = auxiliar; | |  | } | |  |  | |  | float formula (float p1, int p2){ | |  | float auxiliar; | |  |  | |  | intercambio( p1 \* 10, p2); | |  | while(p1 == 10.5 \* p2){ | |  | if( auxiliar < 10.5 and auxiliar >= 8.6 or not auxiliar == 9) then { | |  | auxiliar = auxiliar + -3 + p2 % 045 \* 0-3 / 0+67; | |  |  | |  | auxiliar = auxiliar - (+98 - p1 \* 0x34 % 0xFC4D); | |  | } else { | |  | auxiliar = -3.98 + p2 % 045.27 \* 0-3.11 / 0+67.34; | |  | } | |  | auxiliar = auxiliar - (+98.5 - p1 \* 0x+ABD.EE % 0xFC4D.F4); | |  | } | |  | return auxiliar \* 0x-FF; | |  | } | |  |  | |  | void Main (int args){ | |  | int entero1, entero2; | |  | float real1[10][20], real2; | |  | //Comentario mal cerrado | |  | \\ | |  | for( entero1 = 1 ; entero1 < 21 ; entero1 = entero1 + 1){ | |  | entero2 = 1; | |  | do { | |  | real1[2][10] = 1.0; | |  | entero2 = entero2 + 1; | |  | } until (entero2 == 11) | |  | } | |  | printf('Esto es el programa \'principal\'...'); | |  | { | |  | intercambio(entero2, entero1); | |  | real1[1][1] = formula(real2,entero1 + registro.campo\_a); | |  | } | |  | registro.campo\_b = real1[3][1]; | |  | } | |

Caso error 4:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  | | --- | | struct registro { | |  | int campo\_a; | |  | float campo\_b; | |  | }; | |  |  | |  | void intercambio (int v1, int v2){ | |  | int auxiliar; | |  |  | |  | auxiliar = v1; | |  | v1 = v2; | |  | v2 = auxiliar; | |  | } | |  |  | |  | float formula (float p1, int p2){ | |  | float auxiliar; | |  |  | |  | intercambio( p1 \* 10, p2); | |  | while(p1 == 10.5 \* p2){ | |  | if( auxiliar < 10.5 and auxiliar >= 8.6 or not auxiliar == 9) then { | |  | auxiliar = auxiliar + -3 + p2 % 045 \* 0-3 / 0+67; | |  | //Aqui esta el error | |  | 0x++34 | |  | auxiliar = auxiliar - (+98 - p1 \* 0x34 % 0xFC4D); | |  | } else { | |  | auxiliar = -3.98 + p2 % 045.27 \* 0-3.11 / 0+67.34; | |  | } | |  | auxiliar = auxiliar - (+98.5 - p1 \* 0x+ABD.EE % 0xFC4D.F4); | |  | } | |  | return auxiliar \* 0x-FF; | |  | } | |  |  | |  | void Main (int args){ | |  | int entero1, entero2; | |  | float real1[10][20], real2; | |  | //Aqui esta el error al cerrar el comentario mal | |  | \*/ | |  | for( entero1 = 1 ; entero1 < 21 ; entero1 = entero1 + 1){ | |  | entero2 = 1; | |  | do { | |  | real1[1][20] = 1.0; | |  | entero2 = entero2 + 1; | |  | } until (entero2 == 11) | |  | } | |  | printf('Esto es el programa \'principal\'...'); | |  | { | |  | intercambio(entero2, entero1); | |  | real1[1][1] = formula(real2,entero1 + registro.campo\_a); | |  | } | |  | registro.campo\_b = real1[3][1]; | |  | } | |