

Especificación de un Lenguaje de Programación Mínimo y Construcción de su Procesador (4)

Procesadores de Lenguaje

Restricciones Contextuales

Definición del Lenguaje Fuente

Restricciones Contextuales

1. Especificación de un lenguaje de programación mínimo y construcción de su procesador.

- Introducción a los lenguajes de programación y a los procesadores de lenguajes: el modelo análisis/síntesis.
- Elementos de la teoría de lenguajes formales. Introducción a las gramáticas de atributos: definiciones dirigidas por la sintaxis y semántica de los lenguajes de programación.
- Definición del lenguaje fuente:
 - aspectos léxicos
 - aspectos sintácticos
 - tabla de símbolos
 - **restricciones contextuales**

Definición del Lenguaje Fuente

Restricciones Contextuales

- En la definición de lenguajes de programación suele ser necesario tener en cuenta características contextuales.
- Estas características no son expresables mediante gramáticas incontextuales.
- En el lenguaje ejemplo (el de la 1ª práctica), una restricción contextual es que los identificadores utilizados en las asignaciones y expresiones deben haber sido declarados previamente.
- La situación es análoga a la del siguiente lenguaje
 $\{\alpha\#\alpha \mid \alpha \in \{0,1\}^*\}$
(la primera α es el análogo a la declaración, y la segunda α es análogo al uso) que *no* es incontextual.
- El lenguaje puede *definirse* fácilmente, no obstante, mediante una gramática de atributos.

3

Definición del Lenguaje Fuente

Restricciones Contextuales

- **Idea:**
 - Primero se *relajan* las restricciones contextuales.
 - Después, las restricciones se expresan mediante ecuaciones semánticas.
- **Hecho:** el formalismo de las gramáticas de atributos es *Turing-completo*.
- En el ejemplo el lenguaje *relajado* es $\{\alpha\#\beta \mid \alpha \in \{0,1\}^* \text{ y } \beta \in \{0,1\}^*\}$, que *sí* es incontextual:

```
Sent --> Seq # Seq
Seq  --> Seq 0 | Seq 1 |  $\lambda$ 
```

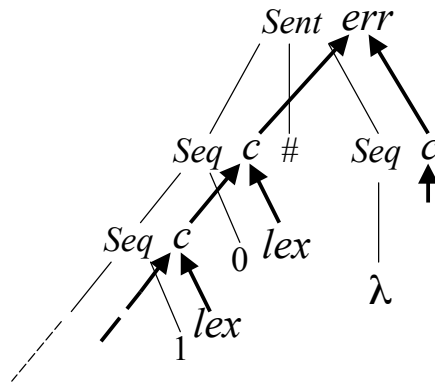
- Después hay que comprobar que $\alpha = \beta$:
 - Se *sintetizan* α y β mediante concatenación de los símbolos que las componen.
 - ...y se comprueba que las cadenas sintetizadas sean iguales.
- *Realización mediante una gramática de atributos*
- *Comparar con lo hecho anteriormente para el lenguaje $a^n b^n c^n$*

4

Definición del Lenguaje Fuente

Restricciones Contextuales

Ejemplo: la frase 101010# no pertenece al lenguaje



5

Definición del Lenguaje Fuente

Restricciones Contextuales

Atributos:

Sent:

sintetizados: err

Seq:

sintetizados: c

Gramática:

Sent --> **Seq** # **Seq** **Sent.err** = (**Seq₀.c** ≠ **Seq₁.c**)

Seq --> **Seq** 0 **Seq₀.c** = **Seq₁.c** || "0"

Seq --> **Seq** 1 **Seq₀.c** = **Seq₁.c** || "1"

Seq --> λ **Seq.c** = ""

- Se descartan todas las cadenas para las cuáles se sintetice *true* para *err*.

6

Definición del Lenguaje Fuente

Restricciones Contextuales

- En el lenguaje ejemplo (práctica 1) las restricciones contextuales se expresarán mediante una gramática de atributos.

ES IMPORTANTE UTILIZAR GRAMÁTICAS DE ATRIBUTOS DISTINTAS

- Para comprobar dichas restricciones se utilizará la información contenida en la tabla de símbolos.
- Se asociará un atributo sintetizado *err* con el axioma. Permitirá indicar si se ha producido o no un error.
- El mecanismo puede refinarse para indicar también información sobre los errores.

7

Definición del Lenguaje Fuente

Restricciones Contextuales

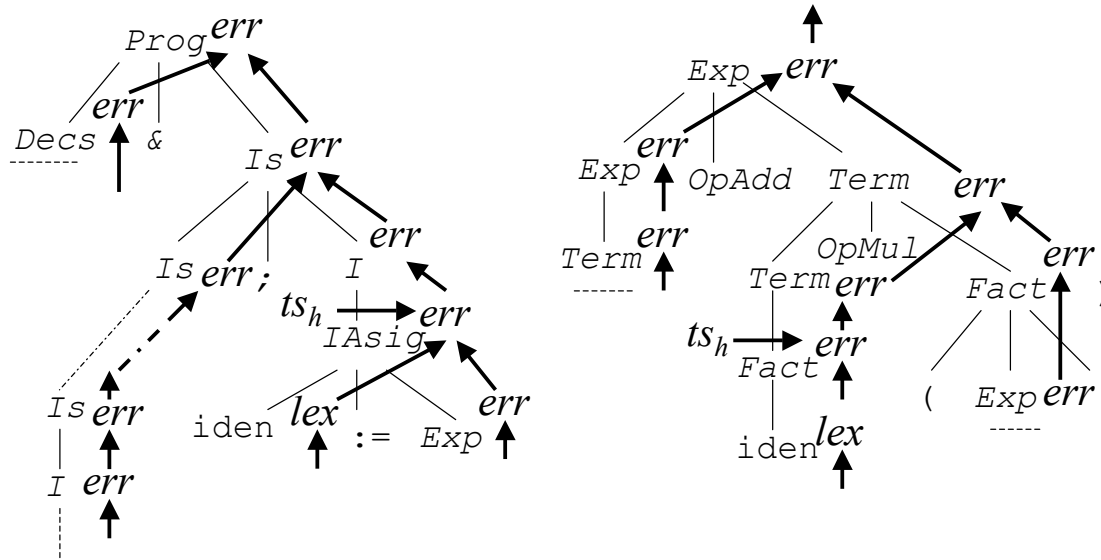
- Algunas restricciones contextuales en el lenguaje ejemplo:
 - Los identificadores utilizados en las instrucciones han debido ser declarados.
 - No se declaran identificadores duplicados.
- Construcción de la gramática de atributos:
 - De acuerdo con Knuth es posible expresar cualquier definición (traducción) dirigida por sintaxis utilizando únicamente atributos sintetizados.
 - Los atributos heredados se introducen como una forma más natural de expresar algunas características del lenguaje (entre ellas las dependencias contextuales).

8

Definición del Lenguaje Fuente

Restricciones Contextuales

Sintetizamos un atributo error para indicar si se utiliza un identificador no declarado o si éste se declara más de una vez



11

Definición del Lenguaje Fuente

Restricciones Contextuales

Atributos:

Prog, Decs, Is, I, IAsig, Exp, Term, Fact: sintetizados: err

Decs: sintetizados: ts, err

iden, Dec: sintetizados: lex

Is, I, IAsig, Exp, Term, Fact: heredados: ts_h

Gramática:

Prog --> *Decs & Is*

Prog.err = *Decs.err* v *Is.err*

Is.ts_h = *Decs.ts*

12

Definición del Lenguaje Fuente

Restricciones Contextuales

```
Decs --> Decs ; Dec
    Decs0.err = Decs1.err ∨
    ¬(obtenerID(Decs.ts, Dec.lex) = null)

Decs --> Dec    ...

Dec --> iden    Dec.lex = iden.lex
```

13

Definición del Lenguaje Fuente

Restricciones Contextuales

```
Is --> I        I.tsh = Is.tsh
                Is.err = I.err

Is --> Is ; I    Is1.tsh = I.tsh = Is0.tsh
                Is0.err = Is1.err ∨ I.err

I --> ISig      ISig.tsh = I.tsh
                I.err = ISig.err

ISig --> iden := Exp
                Exp.tsh = ISig.tsh
                ISig.err =
                    Exp.err ∨
                    (obtenerID(ISig.tsh, iden.lex) = null)
```

14

Definición del Lenguaje Fuente

Restricciones Contextuales

$Exp \rightarrow Exp \text{ OpAd } Term$	$Exp_1.ts_h = Term.ts_h = Exp_o.ts_h$ $Exp_o.err = Exp_1.err \vee Term.err$
$Exp \rightarrow Term$	$Term.ts_h = Exp.ts_h$ $Exp.err = Term.err$
$Term \rightarrow Term \text{ OpMul } Fact$	$Term_1.ts_h = Fact.ts_h = Term_o.ts_h$ $Term_o.err = Term_1.err \vee Fact.err$
$Fact \rightarrow iden$	$Fact.err =$ $(obtenerID(Fact.ts_h, iden.lex) = null)$
$Fact \rightarrow (Exp)$	$Exp.ts_h = Fact.ts_h$ $Fact.err = Exp.err$

15

Definición del Lenguaje Fuente

Restricciones Contextuales

Discusión:

-El lenguaje de la práctica debe incluir al menos dos tipos, lo que implica imponer restricciones a las operaciones que impliquen elementos de diferentes tipos.

- explica como resuelves el problema del acoplamiento entre construcción de la tabla de símbolos y la restricción de que no existan identificadores declarados más de una vez

- Ampliar las restricciones contextuales al caso que el lenguaje permita declarar constantes además de variables.

- Valorar la utilización de atributos remotos en el sentido de Knuth (trasparencias 26 y siguientes del Tema1-2

16