



UNIVERSIDAD DE CÓRDOBA  
ESCUELA POLITÉCNICA SUPERIOR  
DEPARTAMENTO DE INFORMÁTICA Y ANÁLISIS NUMÉRICO

**PROCESADORES DE LENGUAJES**  
GRADO EN INGENIERÍA INFORMÁTICA  
ESPECIALIDAD DE COMPUTACIÓN  
TERCER CURSO, SEGUNDO CUATRIMESTRE



**EJEMPLOS DE GENERACIÓN DE AST**  
**ÁRBOLES SINTAXIS ABSTRACTA**



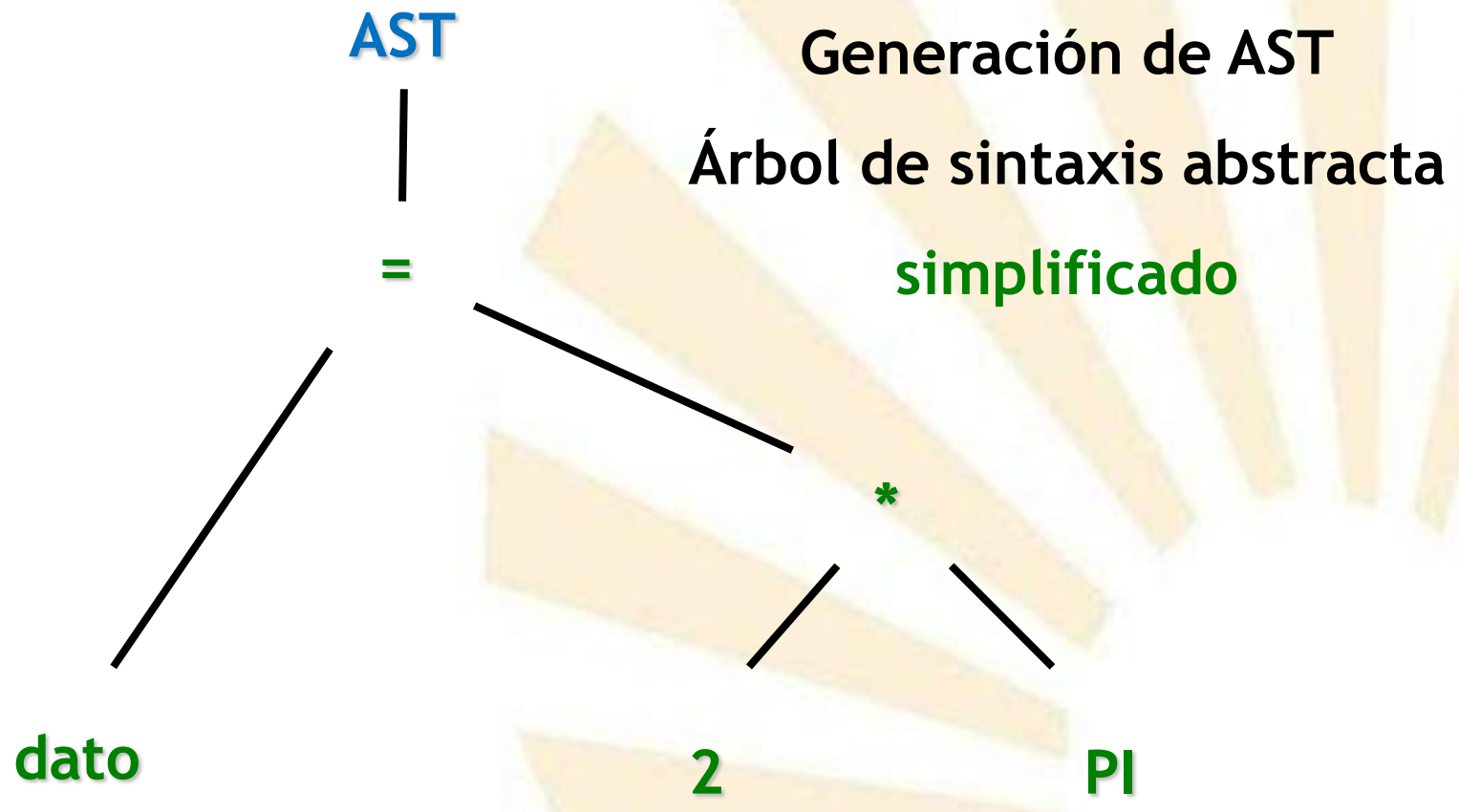
## Ejemplos

- Resumen inicial de ejemplos de AST
- Ejemplo de la gramática 15: `dato = 2 * PI;`
- Ejemplo de la gramática 15: `print sin (PI / 2) ;`
- Ejemplo de la gramática 16: `print (dato > 0) && (dato < 10) ;`
- Ejemplo de la gramática 17: `if (a>0) b = a; else b = -a;`
- Ejemplo de la gramática 17: `while (n>0) { print n; n = n-1;}`

# Ejemplos

- Resumen inicial de ejemplos de AST
- Ejemplo de la gramática 15: `dato = 2 * PI;`
- Ejemplo de la gramática 15: `print sin (PI / 2) ;`
- Ejemplo de la gramática 16: `print (dato > 0) && (dato < 10) ;`
- Ejemplo de la gramática 17: `if (a>0) b = a; else b = -a;`
- Ejemplo de la gramática 17: `while (n>0) { print n; n = n-1;}`

Ejemplo de la gramática 15: dato = 2 \* PI ;

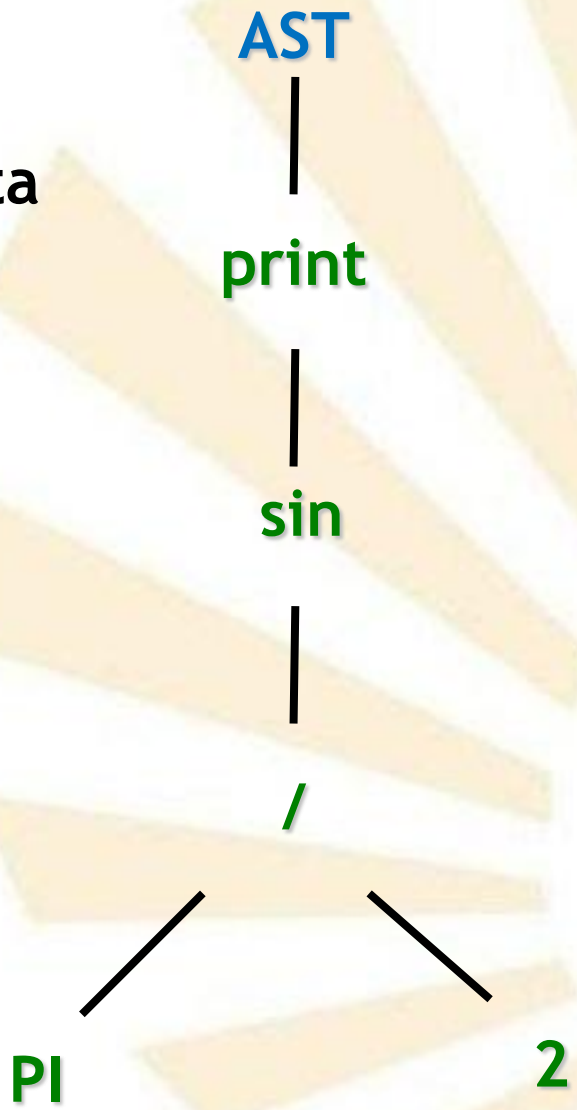


VARIABLE: dato ASSIGNMENT NUMBER: 2 MULT. CONSTANT: PI SEMIC

dato = 2 \* PI ;

# Ejemplo de la gramática 15: `print sin(PI/2);`

AST  
Árbol de sintaxis abstracta  
simplificado

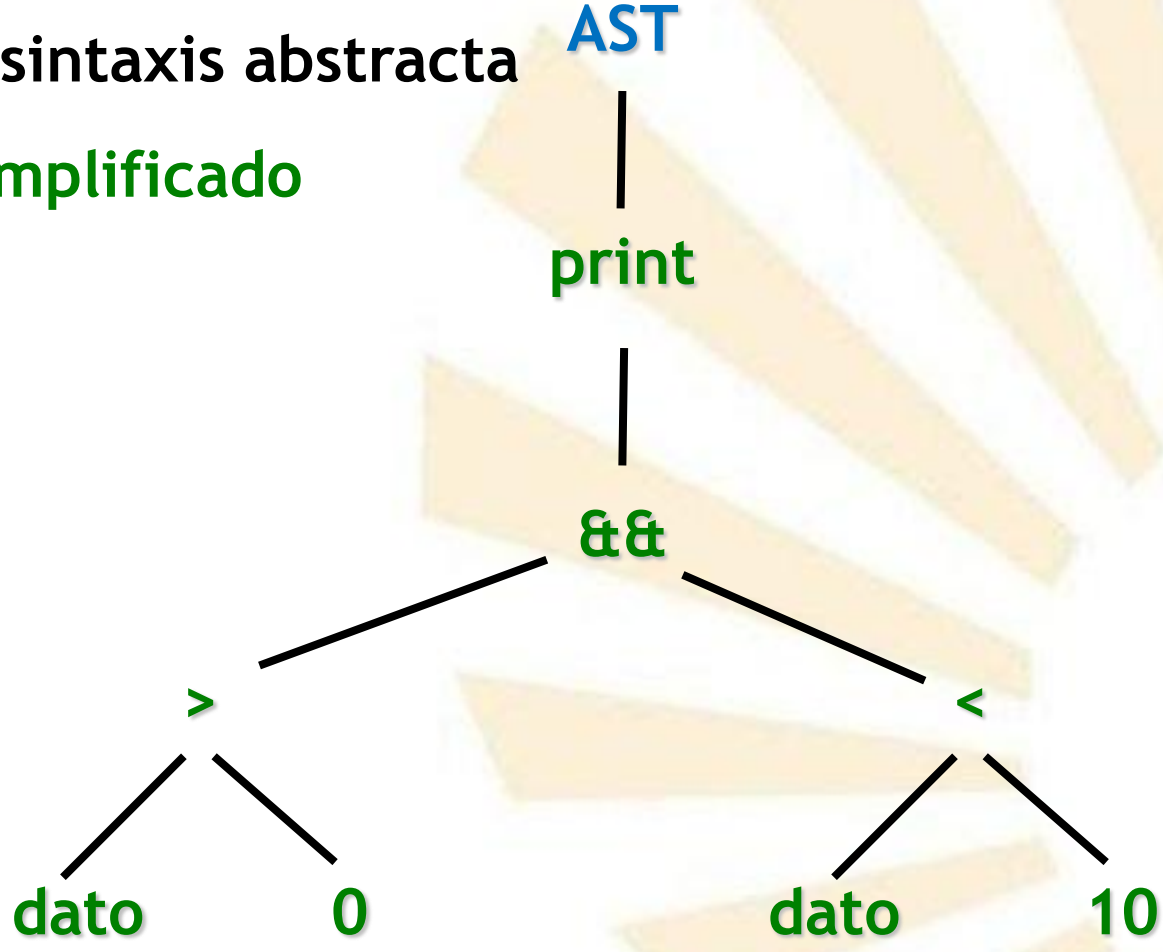


**PRINT BUILTIN: sin LPAREN CONSTANT: PI DIVISION NUMBER: 2 RPAREN SEMIC**  
print sin ( PI / 2 ) ;

# Ejemplo de la gramática 16: `print (dato > 0) && (dato < 10) ;`

AST

Árbol de sintaxis abstracta  
simplificado



```
PRINT LPAREN VAR G_T NUMBER RPAREN AND LPAREN VAR L_T NUMBER RPAREN SEMIC  
print ( dato > 0 ) && ( dato < 10 ) ;
```

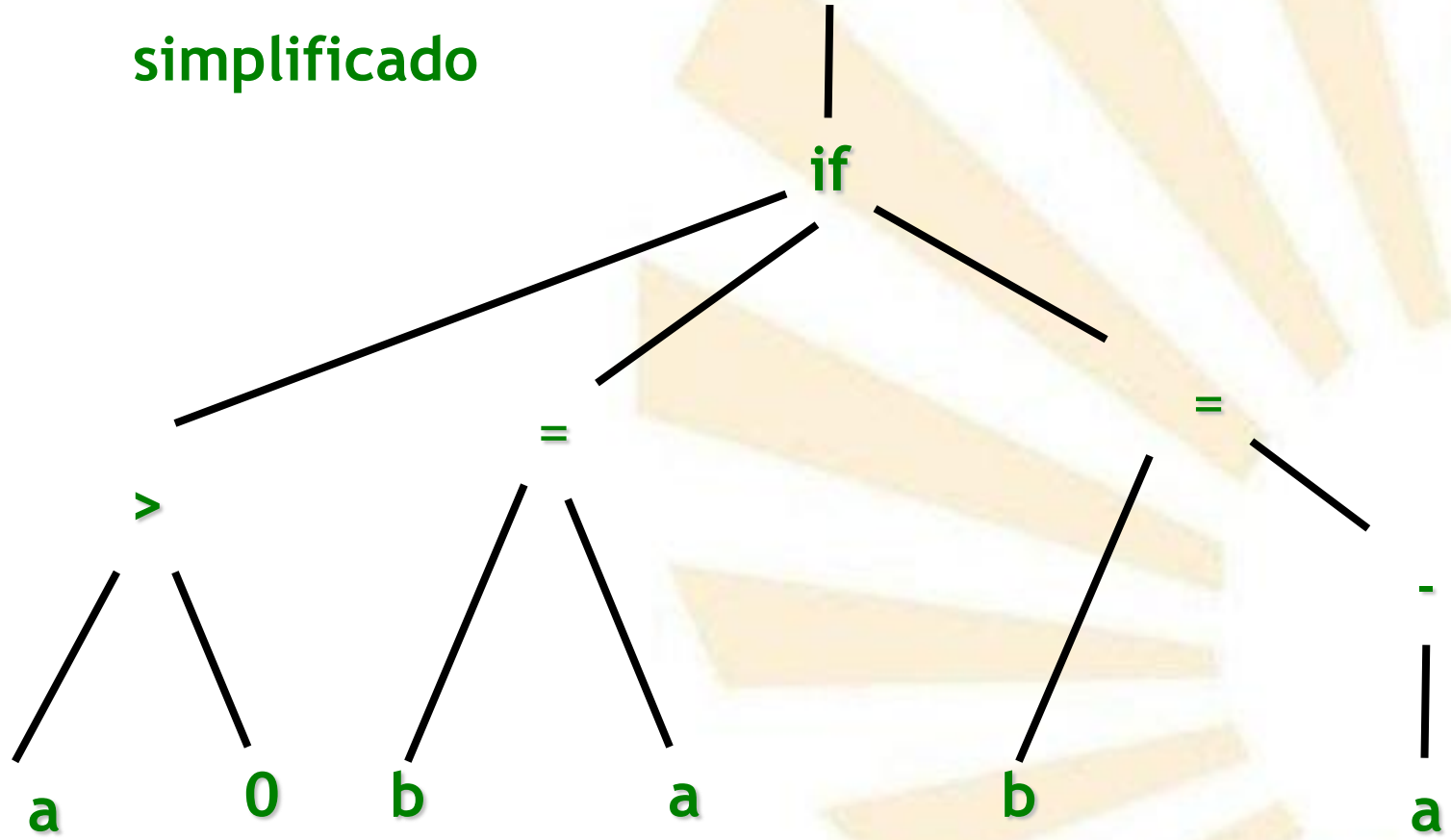
# Ejemplo de la gramática 17: `if (a>0) b = a; else b = -a;`

AST

Árbol de sintaxis abstracta

simplificado

AST



```
IF LP VAR G_T N RP VAR ASSIG VAR SEMIC ELSE VAR ASSIG MINUS VAR SEMIC  
if ( a > 0 ) b = a ; else b = - a ;
```

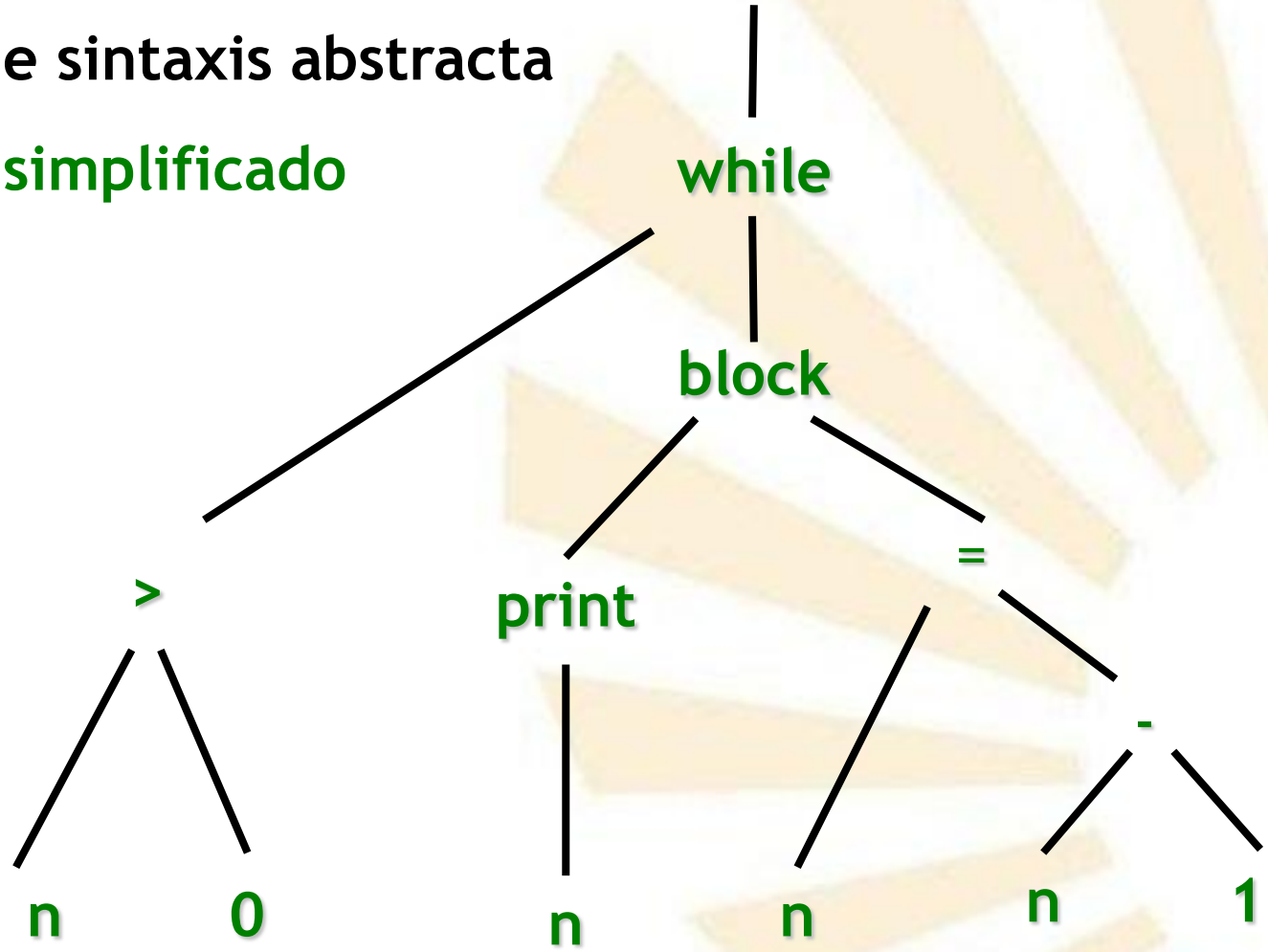


# Ejemplo de la gramática 17: while (n>0) { print n; n = n-1; }

AST

AST

Árbol de sintaxis abstracta  
simplificado



WHILE LP VAR G\_T N RP LB ε PRINT VAR SEMIC VAR ASSIG VAR MINUS N SEMIC RB

while ( n > 0 ) { print n ; n = n - 1 ; }



## Ejemplos

- Resumen inicial de ejemplos de AST
- **Ejemplo de la gramática 15: `dato = 2 * PI;`**
- Ejemplo de la gramática 15: `print sin (PI / 2) ;`
- Ejemplo de la gramática 16: `print (dato > 0) && (dato < 10) ;`
- Ejemplo de la gramática 17: `if (a>0) b = a; else b = -a;`
- Ejemplo de la gramática 17: `while (n>0) { print n; n = n-1;}`

## Ejemplos

- **Ejemplo de la gramática 15:  $\text{dato} = 2 * \text{PI}$ ;**
  - Derivación por la derecha en orden inverso
  - Árbol sintáctico
  - AST: árbol de sintaxis abstracta
  - AST simplificado
  - Construcción “paso a paso” y “sincronizada”

# Ejemplos

- **Ejemplo de la gramática 15:  $\text{dato} = 2 * \text{PI}$ ;**
  - **Derivación por la derecha en orden inverso**
  - **Árbol sintáctico**
  - **AST: árbol de sintaxis abstracta**
  - **AST simplificado**
  - **Construcción “paso a paso” y “sincronizada”**

# Ejemplo de la gramática 15: dato = 2 \* PI ;

Derivación por la derecha en orden inverso

dato = 2 \* PI ;

# Ejemplo de la gramática 15: dato = 2 \* PI ;

Derivación por la derecha en orden inverso

= VARIABLE ASSIGNMENT NUMBER MULTIPLICATION CONSTANT SEMICOLON  
dato = 2 \* PI ;

# Ejemplo de la gramática 15: dato = 2 \* PI ;

Derivación por la derecha en orden inverso

=>  $\epsilon$  VARIABLE ASSIGNMENT NUMBER MULTIPLICATION CONSTANT SEMICOLON  
=  
dato = 2 \* PI ;

# Ejemplo de la gramática 15: dato = 2 \* PI ;

Derivación por la derecha en orden inverso

=> **stmtlist** VARIABLE ASSIGNMENT NUMBER MULTIPLICATION CONSTANT SEMICOLON  
=> **ε** VARIABLE ASSIGNMENT NUMBER MULTIPLICATION CONSTANT SEMICOLON  
=  
**dato = 2 \* PI ;**



# Ejemplo de la gramática 15: dato = 2 \* PI ;

## Derivación por la derecha en orden inverso

=> stmtlist VARIABLE ASSIGNMENT **exp** MULTIPLICATION CONSTANT SEMICOLON  
=> **stmtlist** VARIABLE ASSIGNMENT NUMBER MULTIPLICATION CONSTANT SEMICOLON  
=> **ε** VARIABLE ASSIGNMENT NUMBER MULTIPLICATION CONSTANT SEMICOLON  
=  
**dato**        =        **2**        \*        **PI**        ;

## Ejemplo de la gramática 15: dato = 2 \* PI ;

### Derivación por la derecha en orden inverso

=> stmtlist VARIABLE ASSIGNMENT exp MULTIPLICATION exp SEMICOLON

=> stmtlist VARIABLE ASSIGNMENT exp MULTIPLICATION CONSTANT SEMICOLON

=> stmtlist VARIABLE ASSIGNMENT NUMBER MULTIPLICATION CONSTANT SEMICOLON

=> ε VARIABLE ASSIGNMENT NUMBER MULTIPLICATION CONSTANT SEMICOLON

= VARIABLE ASSIGNMENT NUMBER MULTIPLICATION CONSTANT SEMICOLON

dato = 2 \* PI ;

## Ejemplo de la gramática 15: dato = 2 \* PI ;

### Derivación por la derecha en orden inverso

=> stmtlist VARIABLE ASSIGNMENT **exp** SEMICOLON

=> stmtlist VARIABLE ASSIGNMENT exp MULTIPLICATION **exp** SEMICOLON

=> stmtlist VARIABLE ASSIGNMENT **exp** MULTIPLICATION CONSTANT SEMICOLON

=> **stmtlist** VARIABLE ASSIGNMENT NUMBER MULTIPLICATION CONSTANT SEMICOLON

=> **ε** VARIABLE ASSIGNMENT NUMBER MULTIPLICATION CONSTANT SEMICOLON

= VARIABLE ASSIGNMENT NUMBER MULTIPLICATION CONSTANT SEMICOLON

**dato** = **2** \* **PI** ;

## Ejemplo de la gramática 15: dato = 2 \* PI ;

### Derivación por la derecha en orden inverso

=> stmtlist asgn SEMICOLON

=> stmtlist VARIABLE ASSIGNMENT exp SEMICOLON

=> stmtlist VARIABLE ASSIGNMENT exp MULTIPLICATION exp SEMICOLON

=> stmtlist VARIABLE ASSIGNMENT exp MULTIPLICATION CONSTANT SEMICOLON

=> stmtlist VARIABLE ASSIGNMENT NUMBER MULTIPLICATION CONSTANT SEMICOLON

=> ε VARIABLE ASSIGNMENT NUMBER MULTIPLICATION CONSTANT SEMICOLON

= VARIABLE ASSIGNMENT NUMBER MULTIPLICATION CONSTANT SEMICOLON

dato = 2 \* PI ;

## Ejemplo de la gramática 15: dato = 2 \* PI ;

### Derivación por la derecha en orden inverso

=> stmtlist **stmt**

=> stmtlist asgn SEMICOLON

=> stmtlist VARIABLE ASSIGNMENT **exp** SEMICOLON

=> stmtlist VARIABLE ASSIGNMENT exp MULTIPLICATION **exp** SEMICOLON

=> stmtlist VARIABLE ASSIGNMENT **exp** MULTIPLICATION CONSTANT SEMICOLON

=> **stmtlist** VARIABLE ASSIGNMENT NUMBER MULTIPLICATION CONSTANT SEMICOLON

=> **ε** VARIABLE ASSIGNMENT NUMBER MULTIPLICATION CONSTANT SEMICOLON

= VARIABLE ASSIGNMENT NUMBER MULTIPLICATION CONSTANT SEMICOLON

**dato**        =        **2**        \*        **PI**        ;

# Ejemplo de la gramática 15: dato = 2 \* PI ;

## Derivación por la derecha en orden inverso

=> stmtlist

=> stmtlist stmt

=> stmtlist asgn SEMICOLON

=> stmtlist VARIABLE ASSIGNMENT exp SEMICOLON

=> stmtlist VARIABLE ASSIGNMENT exp MULTIPLICATION exp SEMICOLON

=> stmtlist VARIABLE ASSIGNMENT exp MULTIPLICATION CONSTANT SEMICOLON

=> stmtlist VARIABLE ASSIGNMENT NUMBER MULTIPLICATION CONSTANT SEMICOLON

=> ε VARIABLE ASSIGNMENT NUMBER MULTIPLICATION CONSTANT SEMICOLON

= VARIABLE ASSIGNMENT NUMBER MULTIPLICATION CONSTANT SEMICOLON

dato = 2 \* PI ;

# Ejemplo de la gramática 15: dato = 2 \* PI ;

## Derivación por la derecha en orden inverso

program

=> stmtlist

=> stmtlist stmt

=> stmtlist asgn SEMICOLON

=> stmtlist VARIABLE ASSIGNMENT exp SEMICOLON

=> stmtlist VARIABLE ASSIGNMENT exp MULTIPLICATION exp SEMICOLON

=> stmtlist VARIABLE ASSIGNMENT exp MULTIPLICATION CONSTANT SEMICOLON

=> stmtlist VARIABLE ASSIGNMENT NUMBER MULTIPLICATION CONSTANT SEMICOLON

=> ε VARIABLE ASSIGNMENT NUMBER MULTIPLICATION CONSTANT SEMICOLON

= VARIABLE ASSIGNMENT NUMBER MULTIPLICATION CONSTANT SEMICOLON

dato = 2 \* PI ;



# Ejemplos

- **Ejemplo de la gramática 15:  $\text{dato} = 2 * \text{PI}$ ;**
  - Derivación por la derecha en orden inverso
  - **Árbol sintáctico**
  - **AST: árbol de sintaxis abstracta**
  - **AST simplificado**
  - **Construcción “paso a paso” y “sincronizada”**

**Ejemplo de la gramática 15: dato = 2 \* PI ;**

**Árbol sintáctico**

dato

=

2

\*

PI

;



**Ejemplo de la gramática 15: dato = 2 \* PI ;**

**Árbol sintáctico**

**Análisis léxico**

**VARIABLE: dato ASSIGNMENT NUMBER: 2 MULT. CONSTANT: PI SEMIC**

**dato**

**=**

**2**

**\***

**PI**

**;**

Ejemplo de la gramática 15: dato = 2 \* PI ;

Árbol sintáctico

stmtlist



ε VARIABLE: dato ASSIGNMENT NUMBER: 2 MULT. CONSTANT: PI SEMIC

dato = 2 \* PI ;

# Ejemplo de la gramática 15: dato = 2 \* PI ;

Árbol sintáctico

stmtlist

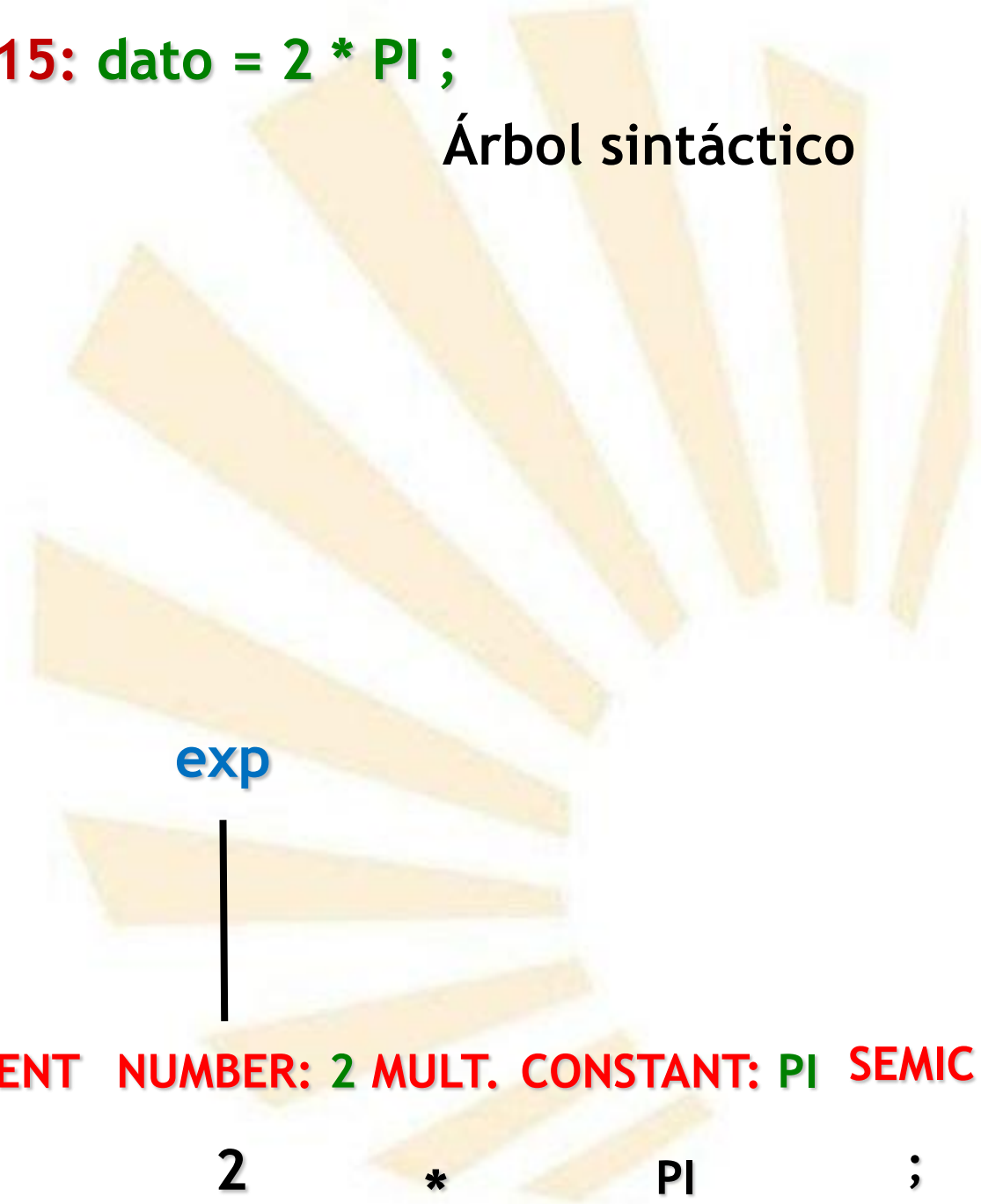


exp



ε VARIABLE: dato ASSIGNMENT NUMBER: 2 MULT. CONSTANT: PI SEMIC

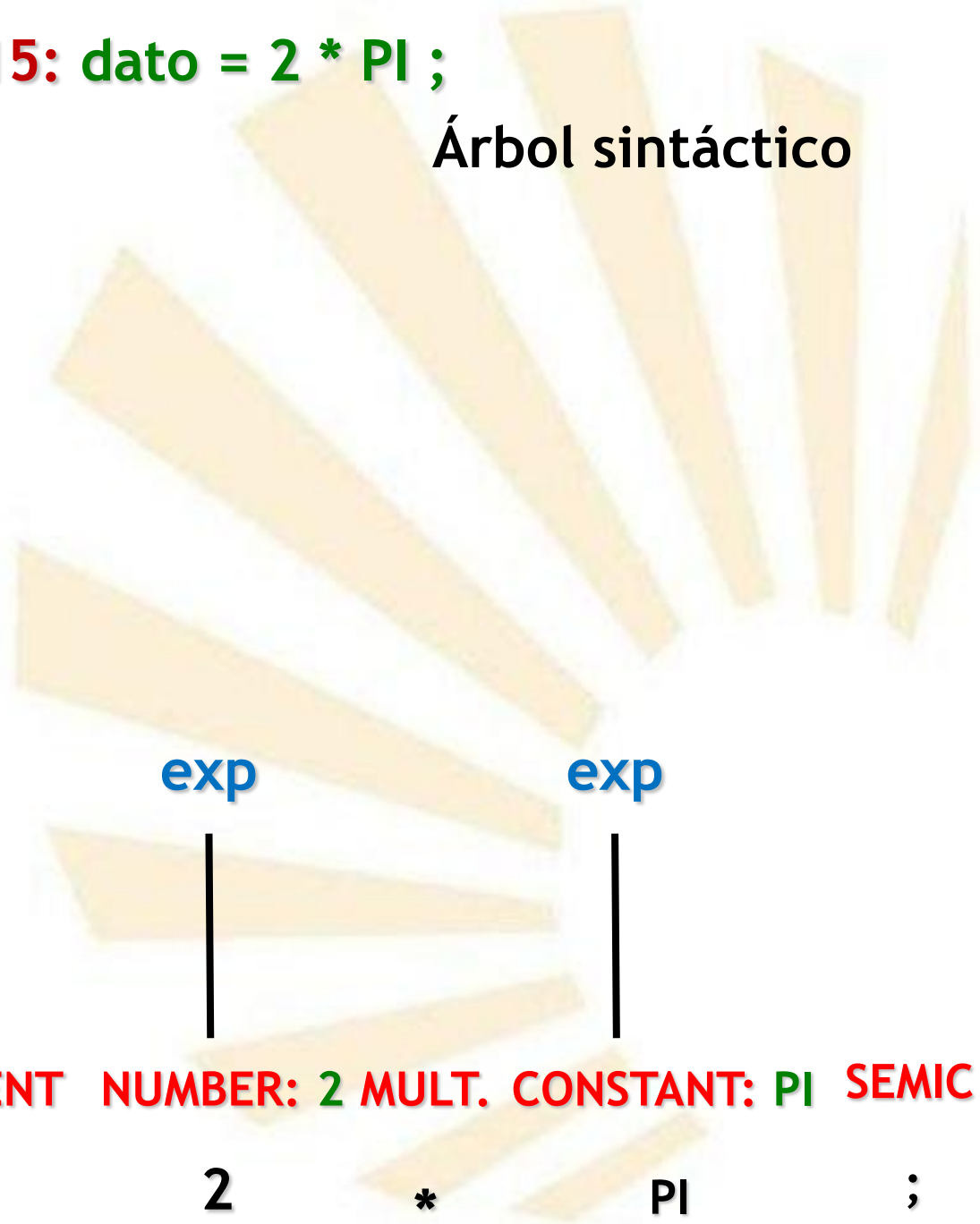
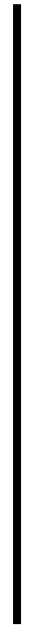
dato = 2 \* PI ;



# Ejemplo de la gramática 15: dato = 2 \* PI ;

Árbol sintáctico

stmtlist



exp

exp



ε VARIABLE: dato ASSIGNMENT = NUMBER: 2 MULT. \* CONSTANT: PI SEMIC ;

dato

=

2

\*

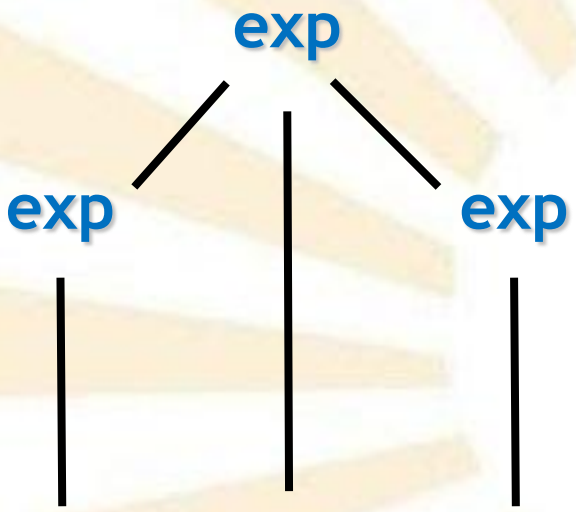
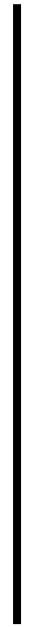
PI

;

# Ejemplo de la gramática 15: dato = 2 \* PI ;

Árbol sintáctico

stmtlist



ε VARIABLE: dato ASSIGNMENT = NUMBER: 2 MULT. \* CONSTANT: PI SEMIC ;

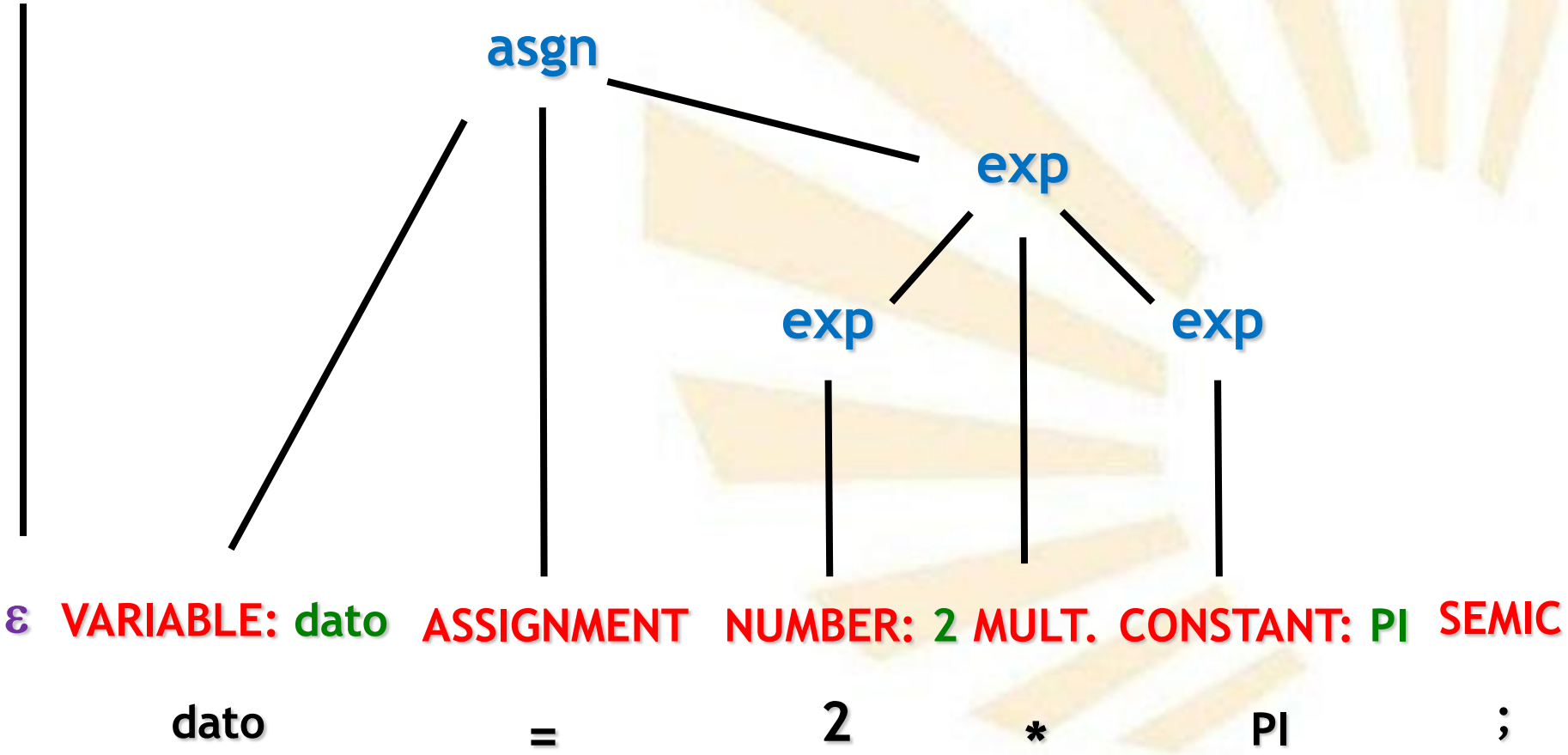
dato = 2 \* PI ;



# Ejemplo de la gramática 15: dato = 2 \* PI ;

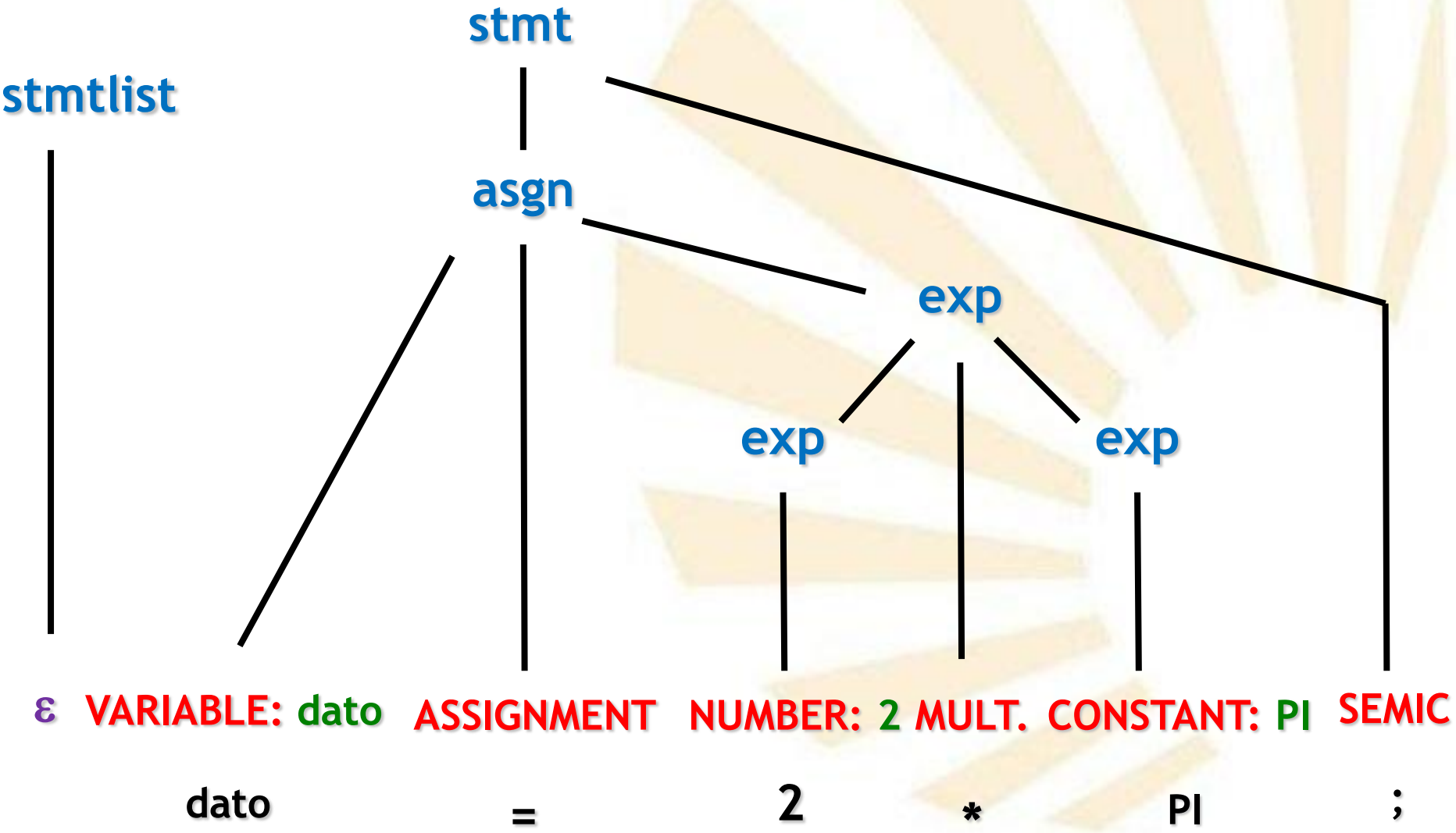
Árbol sintáctico

stmtlist



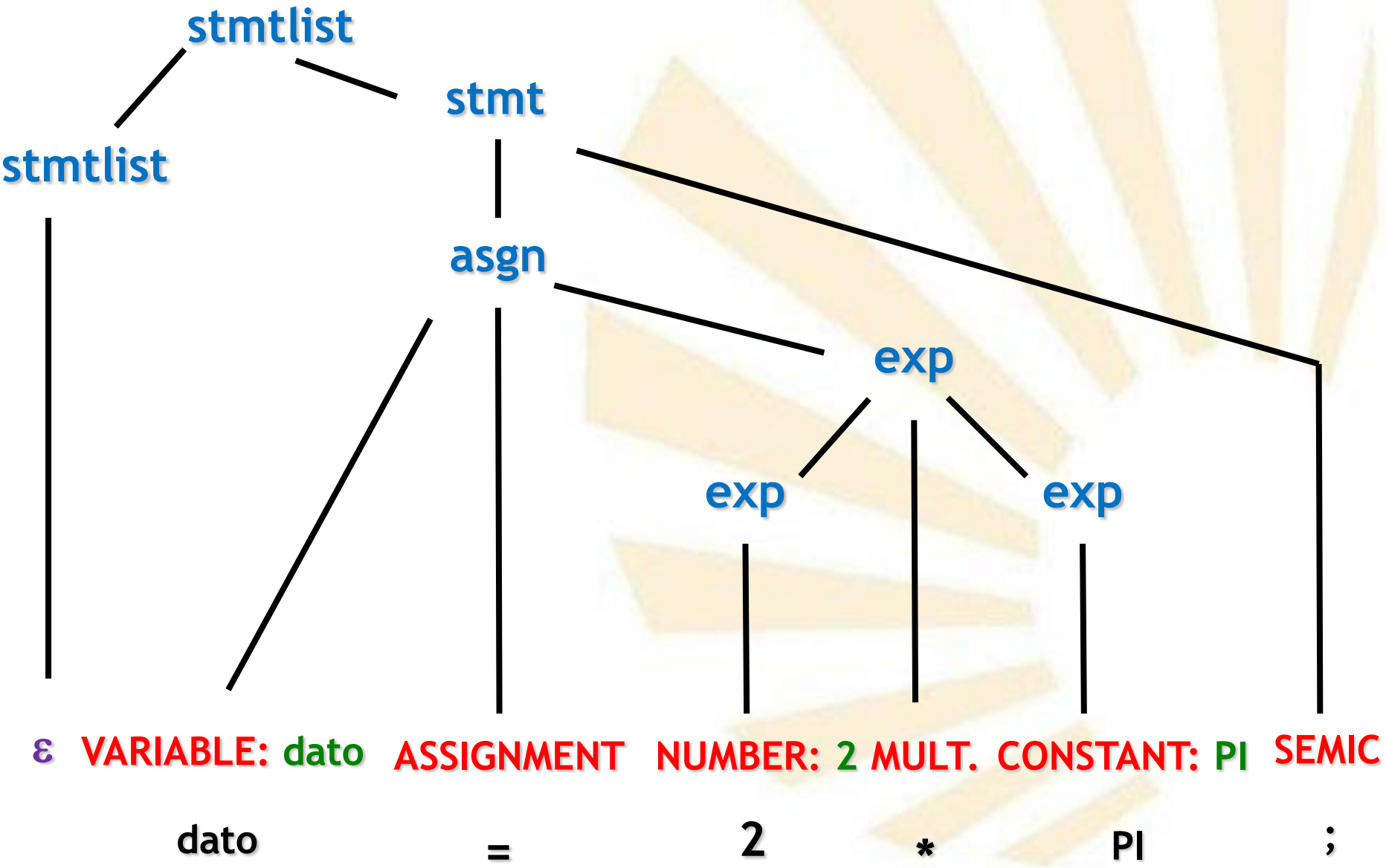
# Ejemplo de la gramática 15: dato = 2 \* PI ;

## Árbol sintáctico



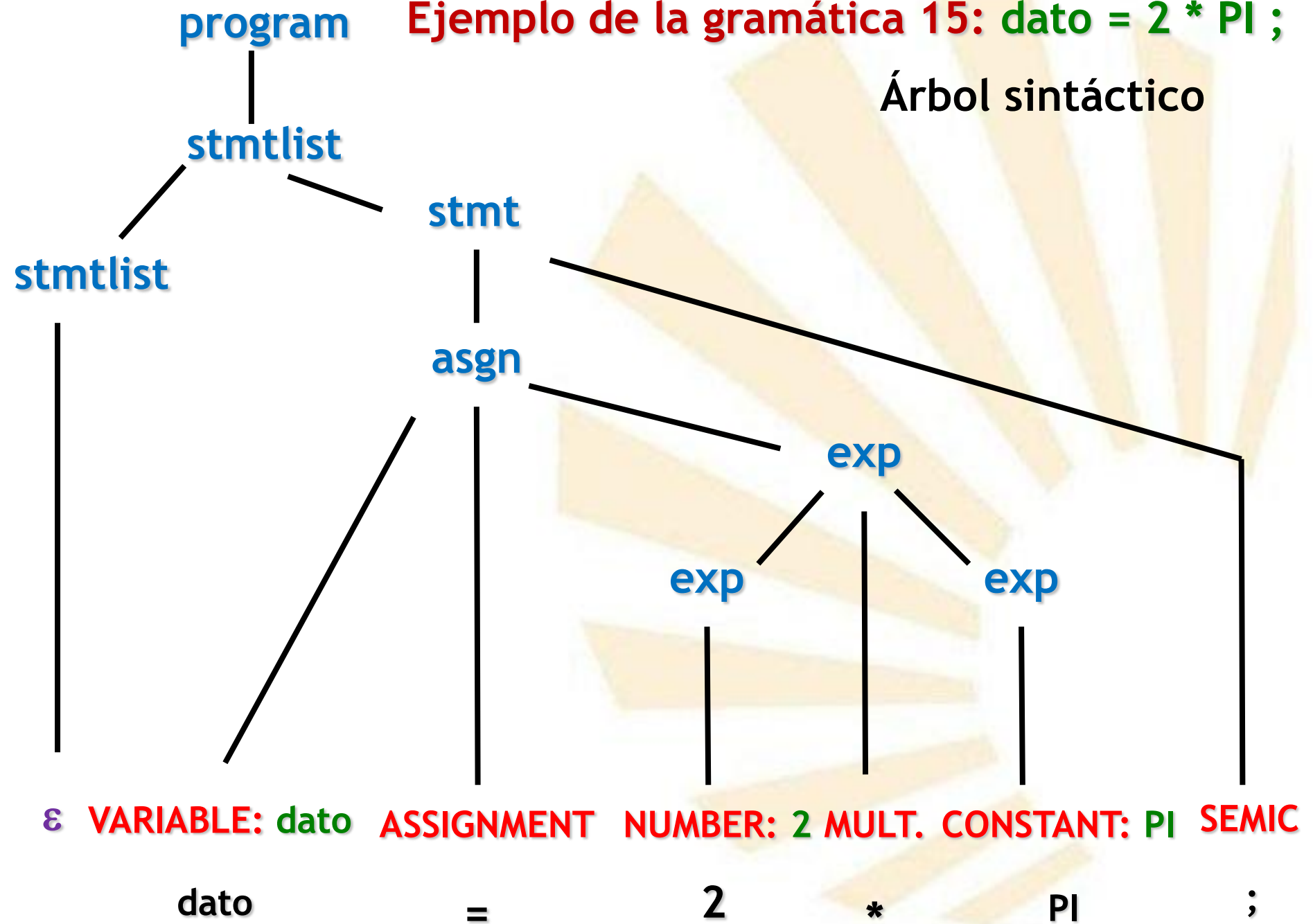
# Ejemplo de la gramática 15: dato = 2 \* PI ;

Árbol sintáctico



# Ejemplo de la gramática 15: dato = 2 \* PI ;

## Árbol sintáctico



## Ejemplos

- **Ejemplo de la gramática 15:  $\text{dato} = 2 * \text{PI}$ ;**
  - Derivación por la derecha en orden inverso
  - Árbol sintáctico
  - **AST: árbol de sintaxis abstracta**
  - AST simplificado
  - Construcción “paso a paso” y “sincronizada”

**Ejemplo de la gramática 15: dato = 2 \* PI ;**

**Generación de AST**

**Árbol de sintaxis abstracta**

**VARIABLE: dato ASSIGNMENT NUMBER: 2 MULT. CONSTANT: PI SEMIC**

**dato**

**=**

**2**

**\***

**PI**

**;**

# Ejemplo de la gramática 15: dato = 2 \* PI ;

Generación de AST

Árbol de sintaxis abstracta

NumberNode



VARIABLE: dato ASSIGNMENT NUMBER: 2 MULT. CONSTANT: PI SEMIC

dato = 2 \* PI ;

# Ejemplo de la gramática 15: dato = 2 \* PI ;

Generación de AST

Árbol de sintaxis abstracta

NumberNode

ConstantNode



VARIABLE: dato ASSIGNMENT NUMBER: 2 MULT. CONSTANT: PI SEMIC

dato = 2 \* PI ;



**Ejemplo de la gramática 15: dato = 2 \* PI ;**

Generación de AST

Árbol de sintaxis abstracta

MultiplicationNode: \*

NumberNode

ConstantNode

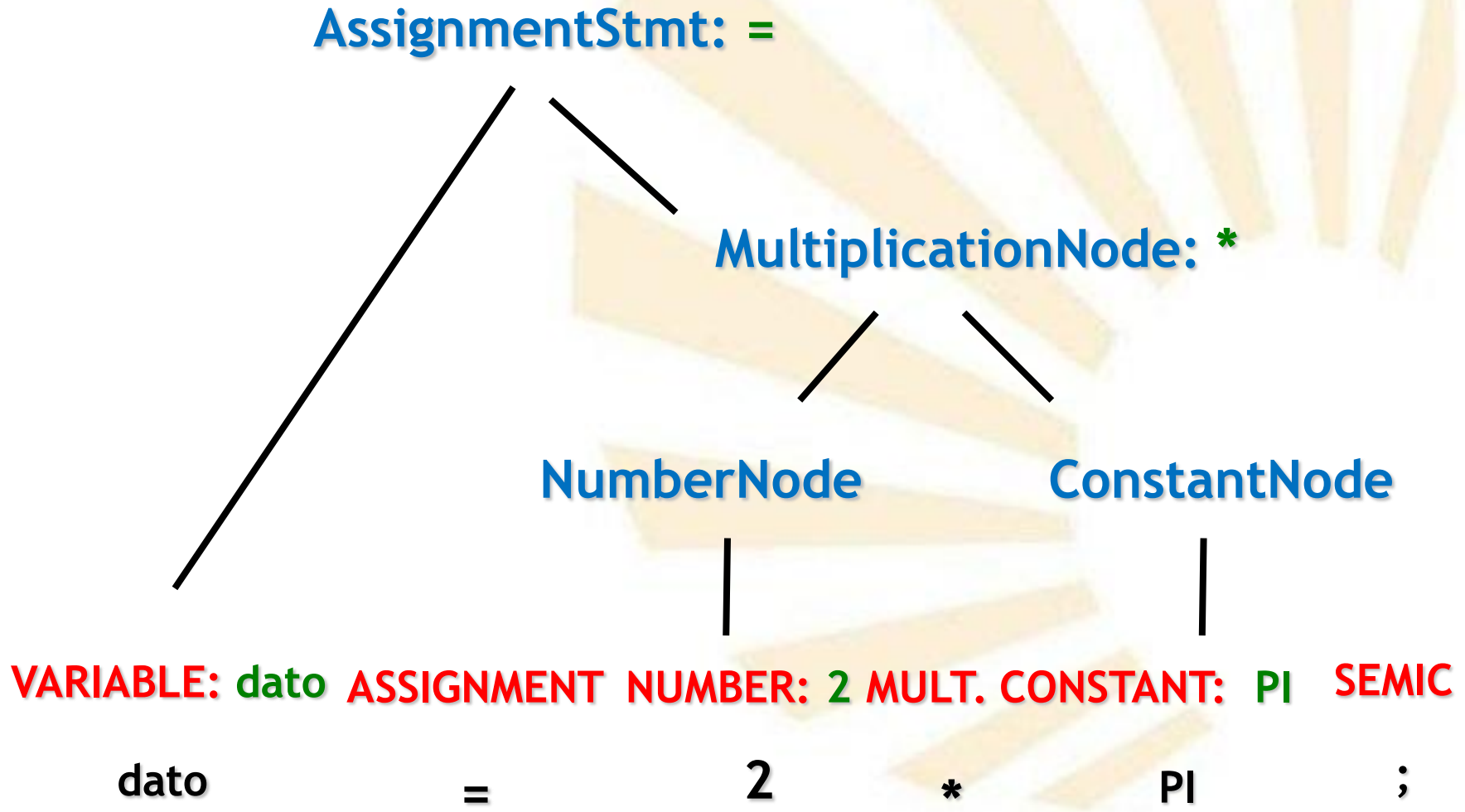
VARIABLE: dato ASSIGNMENT = NUMBER: 2 MULT. \* CONSTANT: PI SEMIC ;

dato = 2 \* PI ;

**Ejemplo de la gramática 15: dato = 2 \* PI ;**

Generación de AST

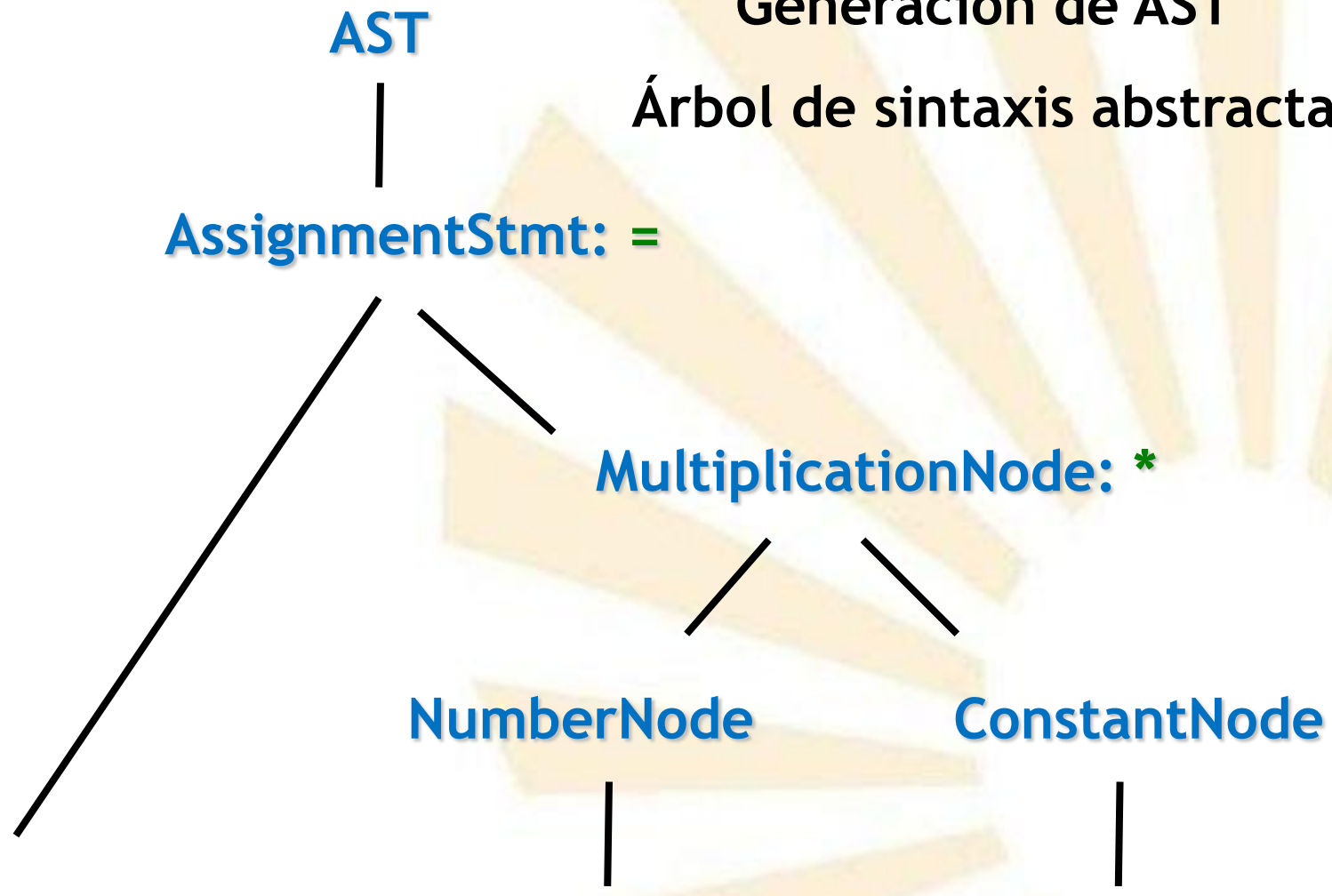
Árbol de sintaxis abstracta



**Ejemplo de la gramática 15: dato = 2 \* PI ;**

Generación de AST

Árbol de sintaxis abstracta



**VARIABLE: dato ASSIGNMENT NUMBER: 2 MULT. CONSTANT: PI SEMIC**

dato = 2 \* PI ;

## Ejemplos

- **Ejemplo de la gramática 15:  $\text{dato} = 2 * \text{PI}$ ;**
  - Derivación por la derecha en orden inverso
  - Árbol sintáctico
  - AST: árbol de sintaxis abstracta
  - **AST simplificado**
  - Construcción “paso a paso” y “sincronizada”

**Ejemplo de la gramática 15: dato = 2 \* PI ;**

**Generación de AST**

**Árbol de sintaxis abstracta  
simplificado**

**2**

**VARIABLE: dato ASSIGNMENT NUMBER: 2 MULT. CONSTANT: PI SEMIC**

**dato**

**=**

**2**

**\***

**PI**

**;**

**Ejemplo de la gramática 15:** dato = 2 \* PI ;

Generación de AST

Árbol de sintaxis abstracta  
simplificado

2

PI

**VARIABLE:** dato **ASSIGNMENT** = **NUMBER:** 2 **MULT.** \* **CONSTANT:** PI **SEMIC** ;

dato

=

2

\*

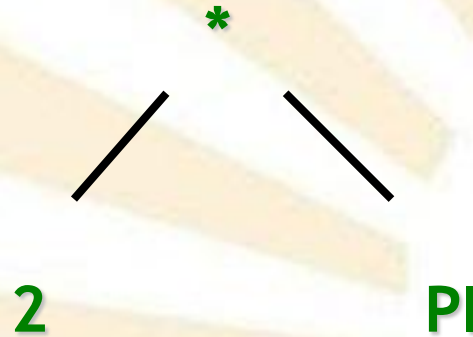
PI

;

Ejemplo de la gramática 15: **dato = 2 \* PI ;**

Generación de AST

Árbol de sintaxis abstracta  
simplificado



**VARIABLE: dato ASSIGNMENT NUMBER: 2 MULT. CONSTANT: PI SEMIC**

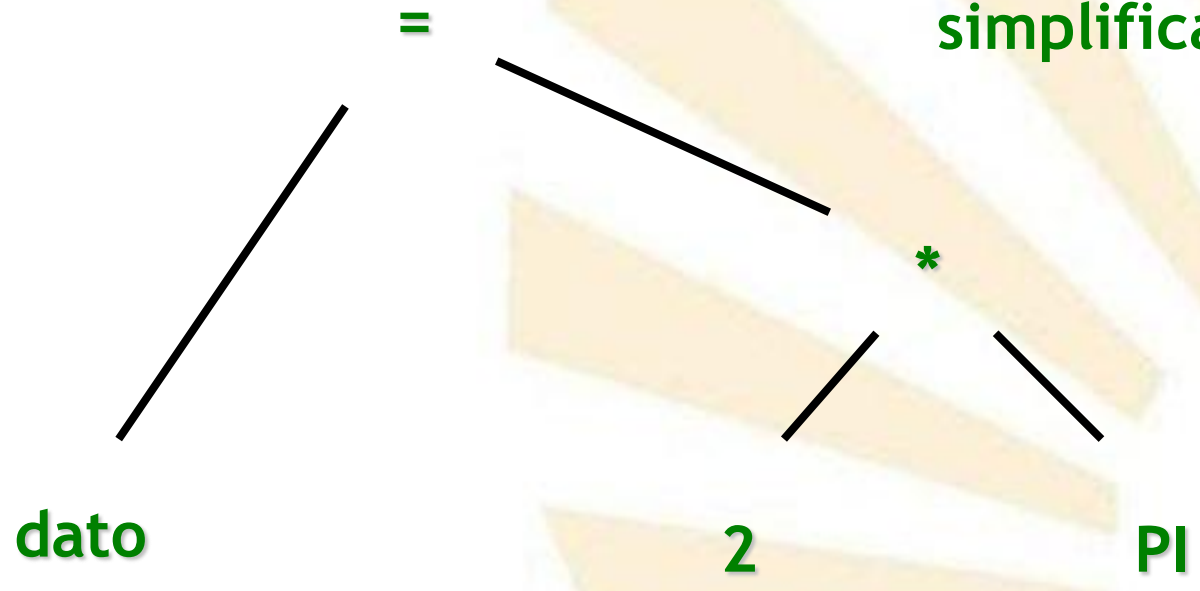
dato = 2 \* PI ;

# Ejemplo de la gramática 15: dato = 2 \* PI ;

Generación de AST

Árbol de sintaxis abstracta

simplificado

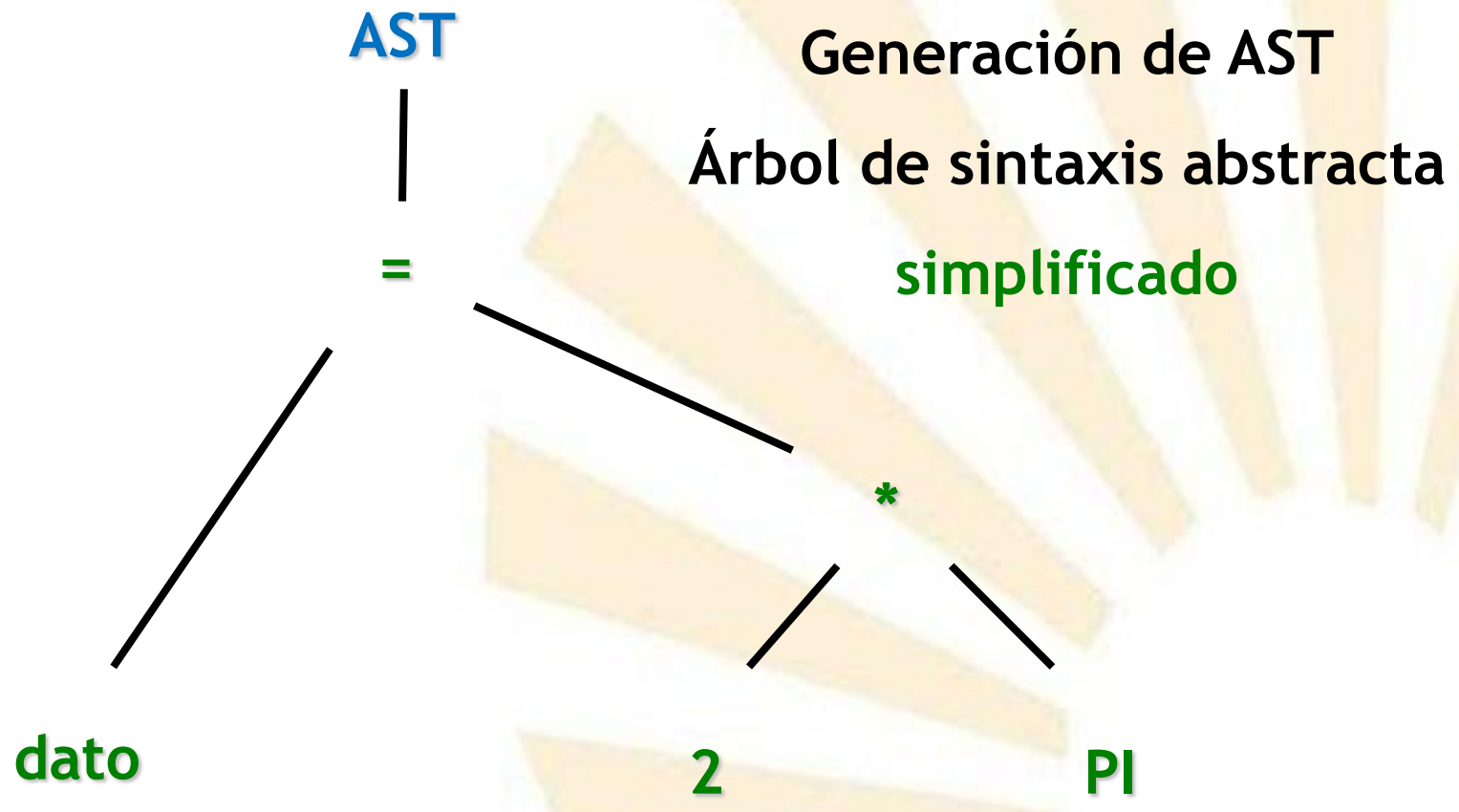


VARIABLE: dato ASSIGNMENT NUMBER: 2 MULT. CONSTANT: PI SEMIC

dato = 2 \* PI ;



Ejemplo de la gramática 15: dato = 2 \* PI ;



VARIABLE: dato ASSIGNMENT NUMBER: 2 MULT. CONSTANT: PI SEMIC

dato = 2 \* PI ;

# Ejemplos

- **Ejemplo de la gramática 15:  $\text{dato} = 2 * \text{PI}$ ;**
  - Derivación por la derecha en orden inverso
  - Árbol sintáctico
  - AST: árbol de sintaxis abstracta
  - AST simplificado
  - Construcción “paso a paso” y “sincronizada”

## Ejemplo de la gramática 15: $\text{dato} = 2 * \text{PI} ;$

Construcción “paso a paso” y “sincronizada” de

- La derivación por la derecha en orden inverso
- El árbol de análisis sintáctico
- El árbol de sintaxis abstracta: AST
- El árbol de sintaxis abstracta simplificado: AST **simplificado**

# Ejemplo de la gramática 15: dato = 2 \* PI ;

Derivación por la derecha en orden inverso

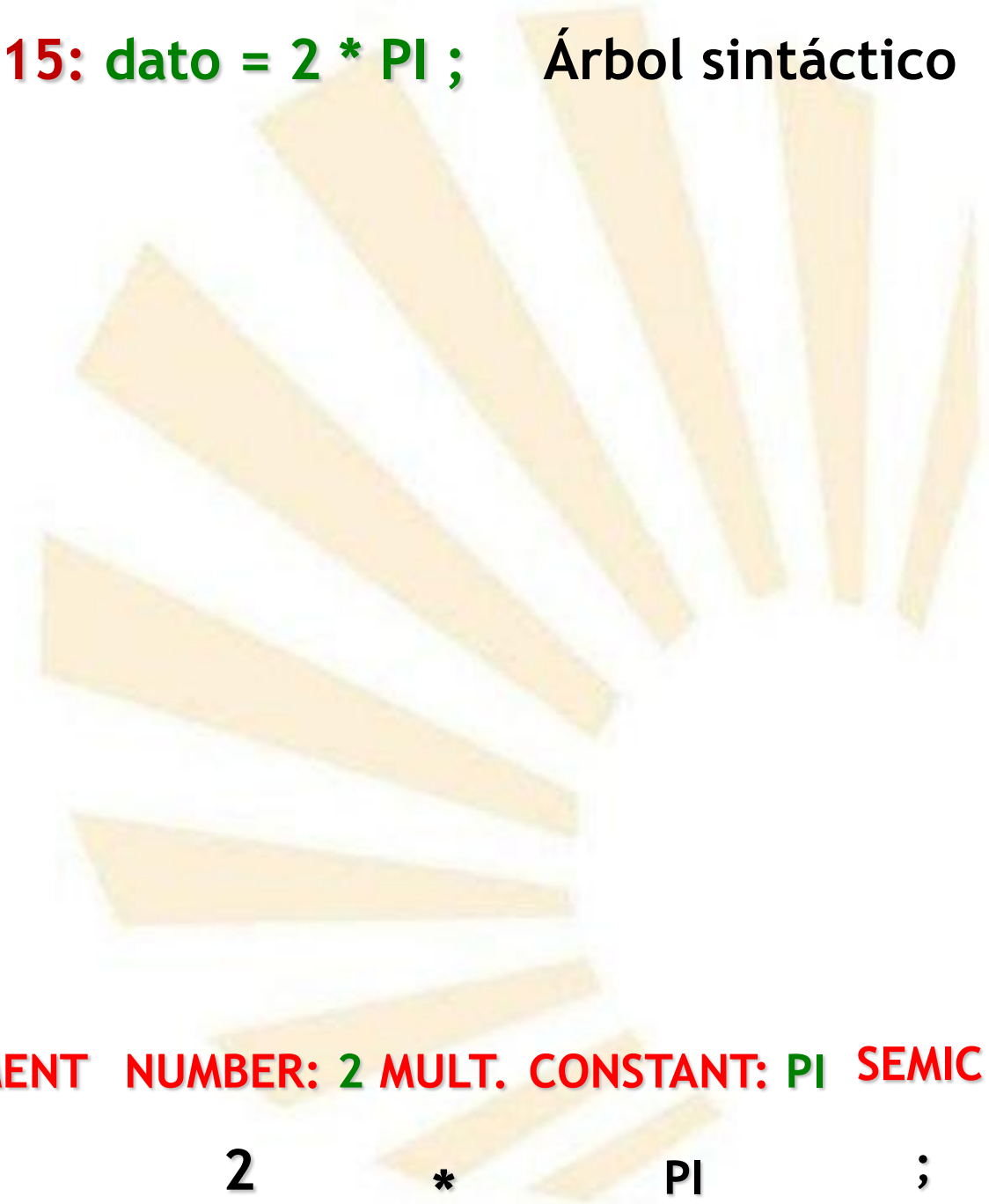
dato = 2 \* PI ;

# Ejemplo de la gramática 15: dato = 2 \* PI ;

Derivación por la derecha en orden inverso

= VARIABLE ASSIGNMENT NUMBER MULTIPLICATION CONSTANT SEMICOLON  
dato = 2 \* PI ;

# Ejemplo de la gramática 15: `dato = 2 * PI ;`    Árbol sintáctico



## Análisis léxico

**VARIABLE:** `dato`    **ASSIGNMENT**    **NUMBER:** `2`    **MULT.**    **CONSTANT:** `PI`    **SEMIC**

`dato`

`=`

`2`

`*`

`PI`

`;`

# Ejemplo de la gramática 15: dato = 2 \* PI ;

Derivación por la derecha en orden inverso

=>  $\epsilon$  VARIABLE ASSIGNMENT NUMBER MULTIPLICATION CONSTANT SEMICOLON  
=  
dato = 2 \* PI ;

# Ejemplo de la gramática 15: dato = 2 \* PI ;

Derivación por la derecha en orden inverso

=> **stmtlist** VARIABLE ASSIGNMENT NUMBER MULTIPLICATION CONSTANT SEMICOLON  
=> **ε** VARIABLE ASSIGNMENT NUMBER MULTIPLICATION CONSTANT SEMICOLON  
=  
**dato = 2 \* PI ;**



# Ejemplo de la gramática 15: dato = 2 \* PI ;    Árbol sintáctico

stmtlist



ε VARIABLE: dato ASSIGNMENT NUMBER: 2 MULT. CONSTANT: PI SEMIC

dato                    =                    2                    \*                    PI                    ;

# Ejemplo de la gramática 15: dato = 2 \* PI ;

## Derivación por la derecha en orden inverso

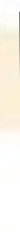
=> stmtlist VARIABLE ASSIGNMENT **exp** MULTIPLICATION CONSTANT SEMICOLON  
=> **stmtlist** VARIABLE ASSIGNMENT NUMBER MULTIPLICATION CONSTANT SEMICOLON  
=> **ε** VARIABLE ASSIGNMENT NUMBER MULTIPLICATION CONSTANT SEMICOLON  
=  
**dato**        =        **2**        \*        **PI**        ;

# Ejemplo de la gramática 15: dato = 2 \* PI ;    Árbol sintáctico

stmtlist

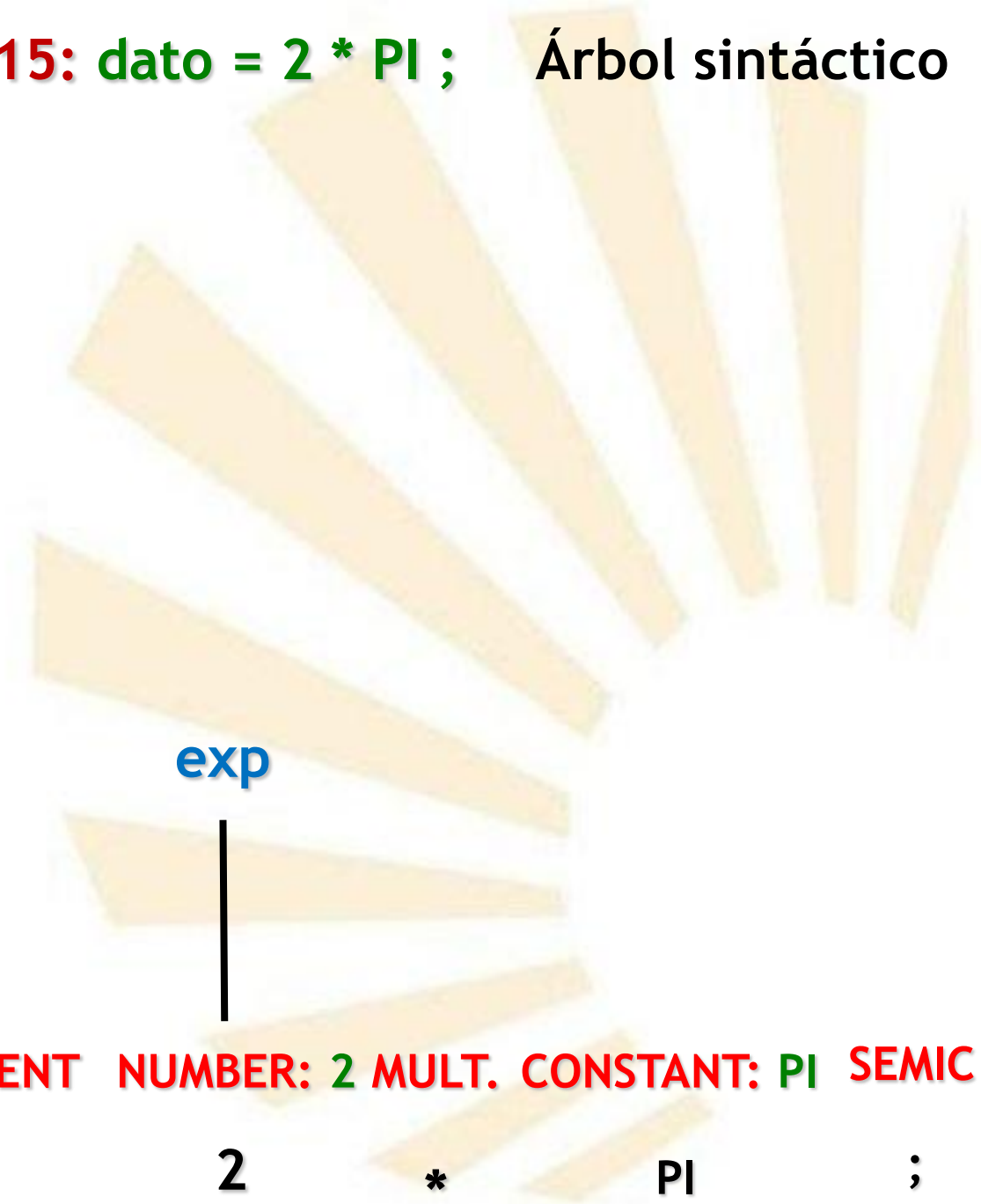


exp



ε VARIABLE: dato ASSIGNMENT NUMBER: 2 MULT. CONSTANT: PI SEMIC

dato = 2 \* PI ;



# Ejemplo de la gramática 15: dato = 2 \* PI ;

NumberNode



VARIABLE: dato ASSIGNMENT NUMBER: 2 MULT. CONSTANT: PI SEMIC

```
dato = 2 * PI ;
```

**Ejemplo de la gramática 15: dato = 2 \* PI ;**

**Generación de AST**

**Árbol de sintaxis abstracta  
simplificado**

**2**

**VARIABLE: dato ASSIGNMENT NUMBER: 2 MULT. CONSTANT: PI SEMIC**

**dato**

**=**

**2**

**\***

**PI**

**;**

## Ejemplo de la gramática 15: dato = 2 \* PI ;

### Derivación por la derecha en orden inverso

=> stmtlist VARIABLE ASSIGNMENT exp MULTIPLICATION **exp** SEMICOLON

=> stmtlist VARIABLE ASSIGNMENT **exp** MULTIPLICATION CONSTANT SEMICOLON

=> **stmtlist** VARIABLE ASSIGNMENT NUMBER MULTIPLICATION CONSTANT SEMICOLON

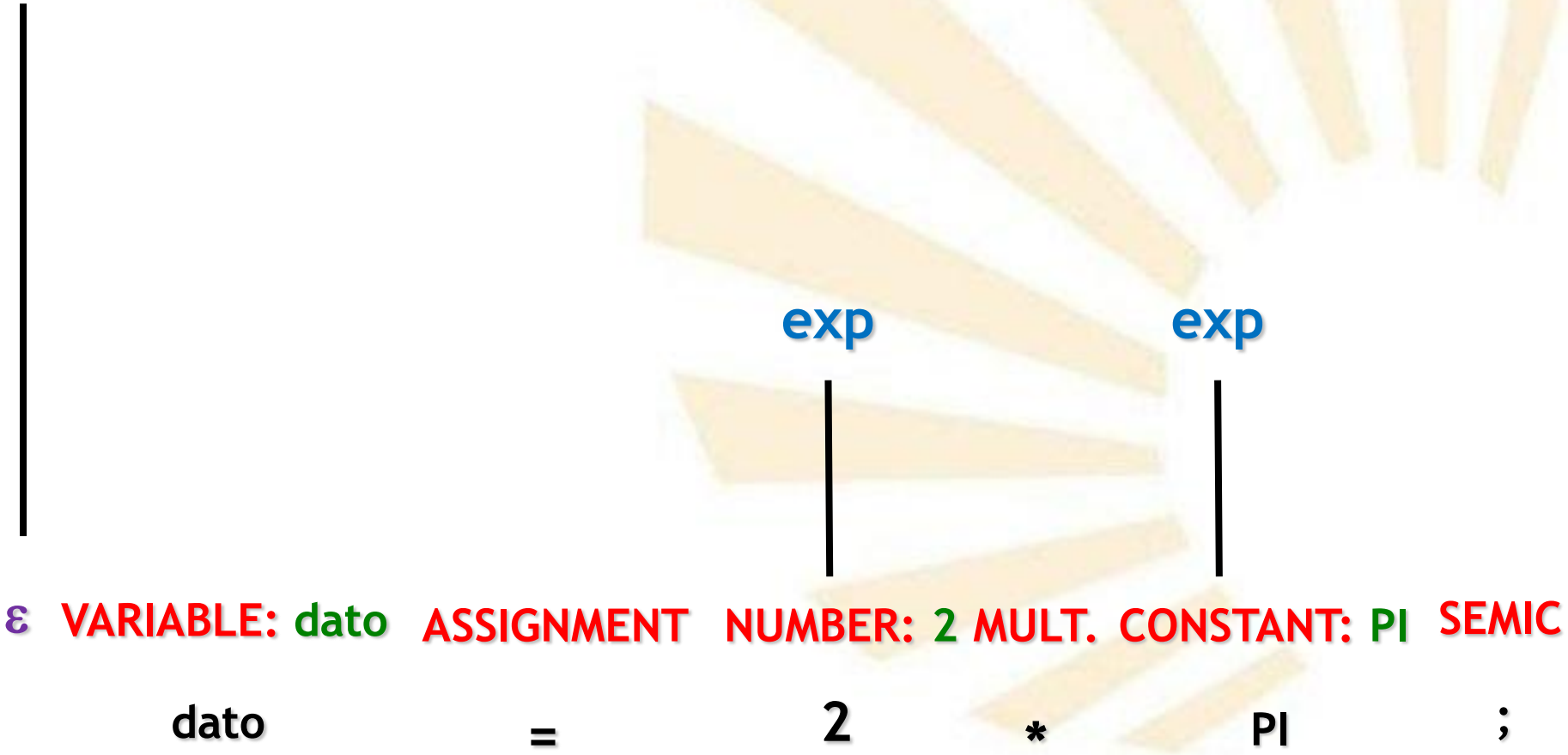
=> **ε** VARIABLE ASSIGNMENT NUMBER MULTIPLICATION CONSTANT SEMICOLON

= VARIABLE ASSIGNMENT NUMBER MULTIPLICATION CONSTANT SEMICOLON

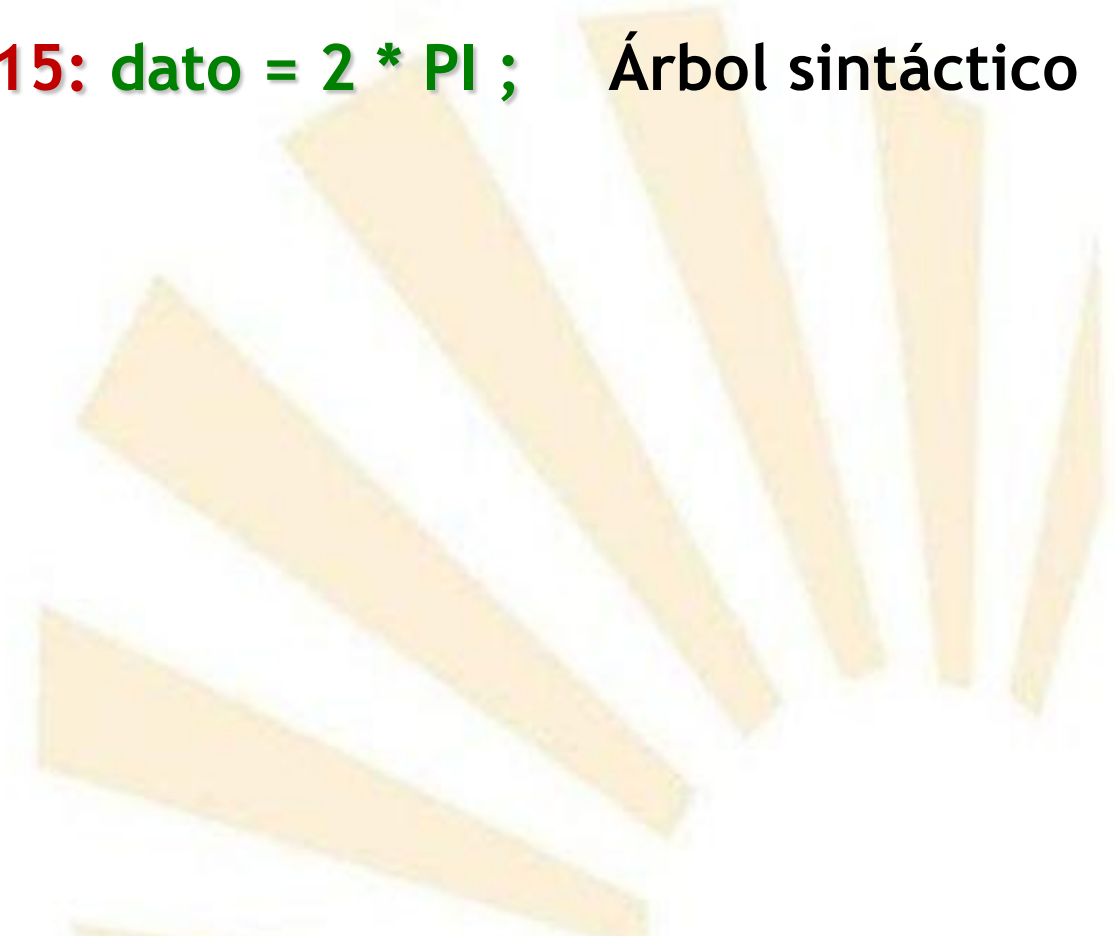
**dato**        =        **2**        \*        **PI**        ;

# Ejemplo de la gramática 15: dato = 2 \* PI ;    Árbol sintáctico

stmtlist



# Ejemplo de la gramática 15: dato = 2 \* PI ;    Árbol sintáctico



NumberNode

ConstantNode



VARIABLE: dato ASSIGNMENT = NUMBER: 2 MULT. \* CONSTANT: PI SEMIC ;

dato

=

2

\*

PI

;



**Ejemplo de la gramática 15: dato = 2 \* PI ;**

**Generación de AST**

**Árbol de sintaxis abstracta  
simplificado**

**2**

**PI**

**VARIABLE: dato ASSIGNMENT NUMBER: 2 MULT. CONSTANT: PI SEMIC**

**dato**

**=**

**2**

**\***

**PI**

**;**

## Ejemplo de la gramática 15: dato = 2 \* PI ;

### Derivación por la derecha en orden inverso

=> stmtlist VARIABLE ASSIGNMENT **exp** SEMICOLON

=> stmtlist VARIABLE ASSIGNMENT exp MULTIPLICATION **exp** SEMICOLON

=> stmtlist VARIABLE ASSIGNMENT **exp** MULTIPLICATION CONSTANT SEMICOLON

=> **stmtlist** VARIABLE ASSIGNMENT NUMBER MULTIPLICATION CONSTANT SEMICOLON

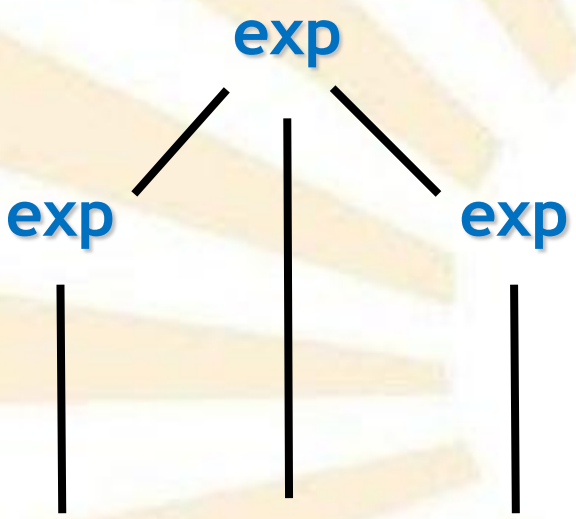
=> **ε** VARIABLE ASSIGNMENT NUMBER MULTIPLICATION CONSTANT SEMICOLON

= VARIABLE ASSIGNMENT NUMBER MULTIPLICATION CONSTANT SEMICOLON

**dato** = **2** \* **PI** ;

# Ejemplo de la gramática 15: dato = 2 \* PI ;    Árbol sintáctico

stmtlist



ε VARIABLE: dato ASSIGNMENT NUMBER: 2 MULT. CONSTANT: PI SEMIC

dato = 2 \* PI ;

# Ejemplo de la gramática 15: dato = 2 \* PI ;

AST de \*

MultiplicationNode: \*

NumberNode

ConstantNode

VARIABLE: dato ASSIGNMENT = NUMBER: 2 MULT. \* CONSTANT: PI SEMIC ;

dato

=

2

\*

PI

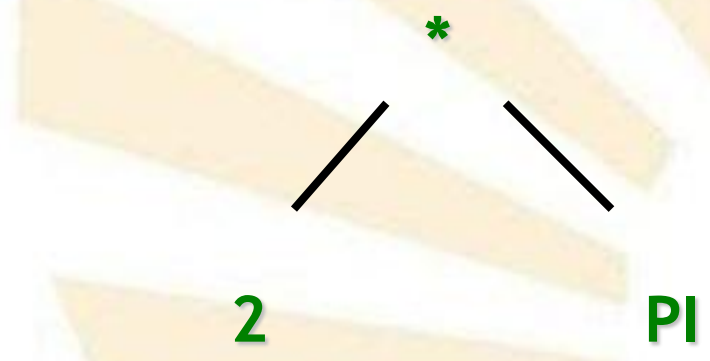
;

Ejemplo de la gramática 15: dato = 2 \* PI ;

Generación de AST

Árbol de sintaxis abstracta simplificado

AST de \*



VARIABLE: dato ASSIGNMENT NUMBER: 2 MULT. CONSTANT: PI SEMIC

dato = 2 \* PI ;

## Ejemplo de la gramática 15: dato = 2 \* PI ;

### Derivación por la derecha en orden inverso

=> stmtlist asgn SEMICOLON

=> stmtlist VARIABLE ASSIGNMENT exp SEMICOLON

=> stmtlist VARIABLE ASSIGNMENT exp MULTIPLICATION exp SEMICOLON

=> stmtlist VARIABLE ASSIGNMENT exp MULTIPLICATION CONSTANT SEMICOLON

=> stmtlist VARIABLE ASSIGNMENT NUMBER MULTIPLICATION CONSTANT SEMICOLON

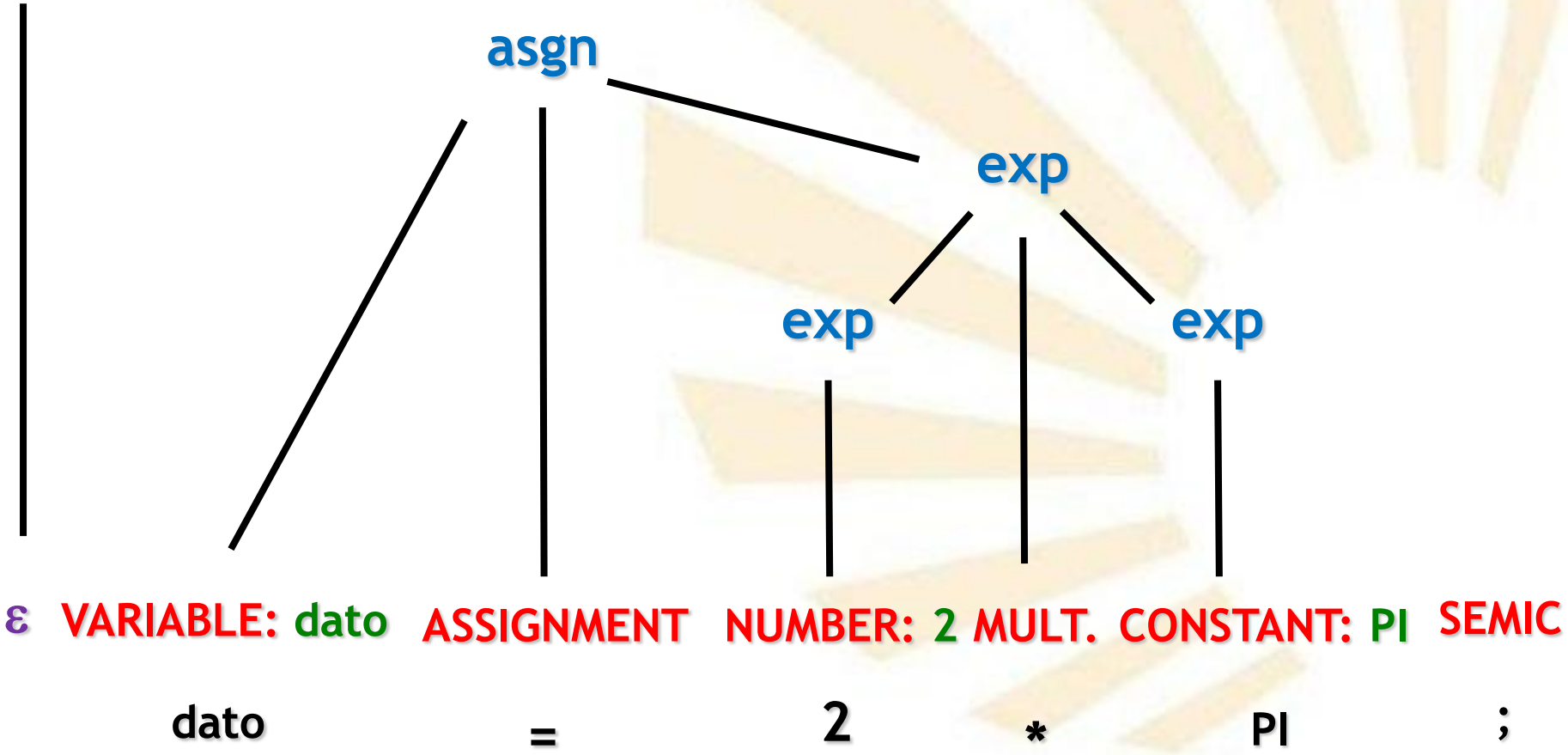
=> ε VARIABLE ASSIGNMENT NUMBER MULTIPLICATION CONSTANT SEMICOLON

= VARIABLE ASSIGNMENT NUMBER MULTIPLICATION CONSTANT SEMICOLON

dato = 2 \* PI ;

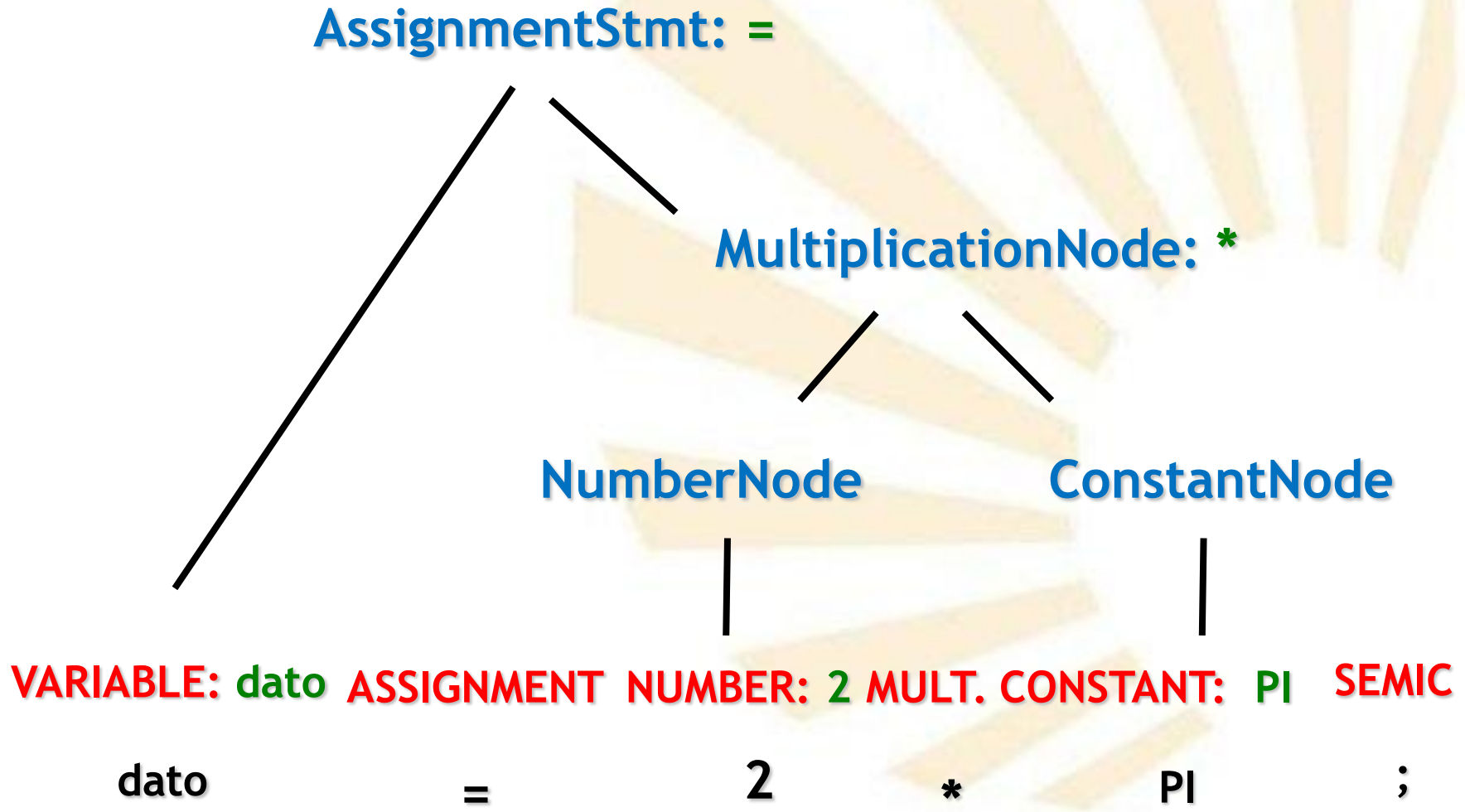
# Ejemplo de la gramática 15: dato = 2 \* PI ;    Árbol sintáctico

stmtlist



# Ejemplo de la gramática 15: dato = 2 \* PI ;

AST de asgn

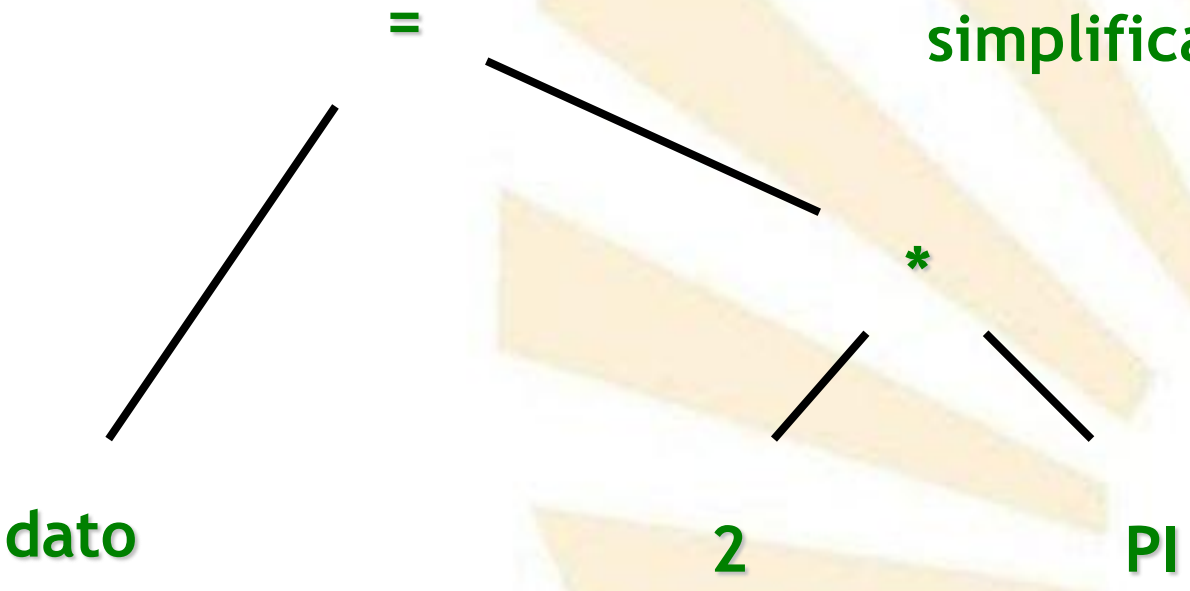




# Ejemplo de la gramática 15: dato = 2 \* PI ;

AST de asgn

Generación de AST  
Árbol de sintaxis abstracta  
simplificado



VARIABLE: dato ASSIGNMENT NUMBER: 2 MULT. CONSTANT: PI SEMIC

dato = 2 \* PI ;

## Ejemplo de la gramática 15: dato = 2 \* PI ;

### Derivación por la derecha en orden inverso

=> stmtlist **stmt**

=> stmtlist **asgn** SEMICOLON

=> stmtlist VARIABLE ASSIGNMENT **exp** SEMICOLON

=> stmtlist VARIABLE ASSIGNMENT exp MULTIPLICATION **exp** SEMICOLON

=> stmtlist VARIABLE ASSIGNMENT **exp** MULTIPLICATION CONSTANT SEMICOLON

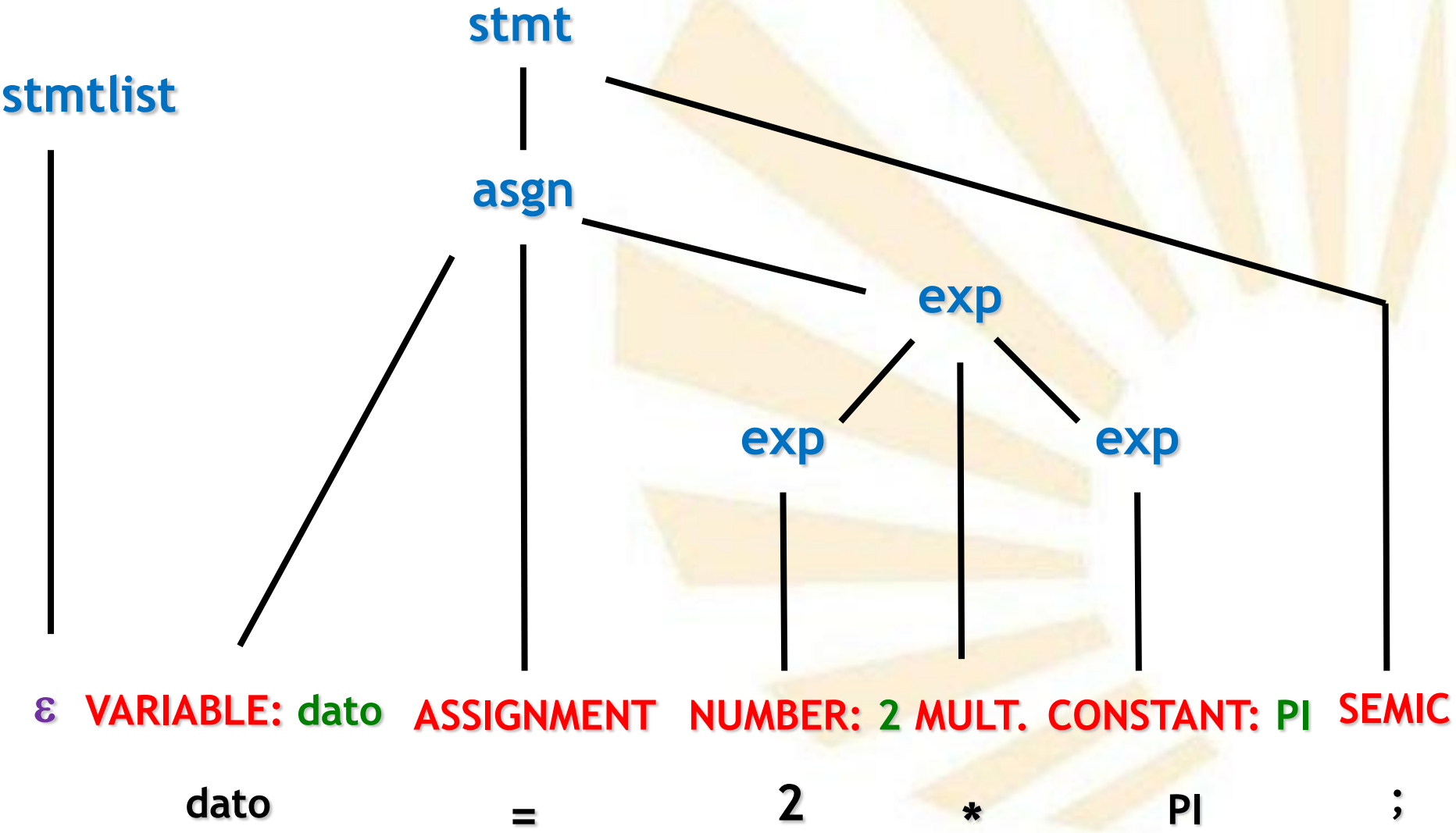
=> **stmtlist** VARIABLE ASSIGNMENT NUMBER MULTIPLICATION CONSTANT SEMICOLON

=> **ε** VARIABLE ASSIGNMENT NUMBER MULTIPLICATION CONSTANT SEMICOLON

= VARIABLE ASSIGNMENT NUMBER MULTIPLICATION CONSTANT SEMICOLON

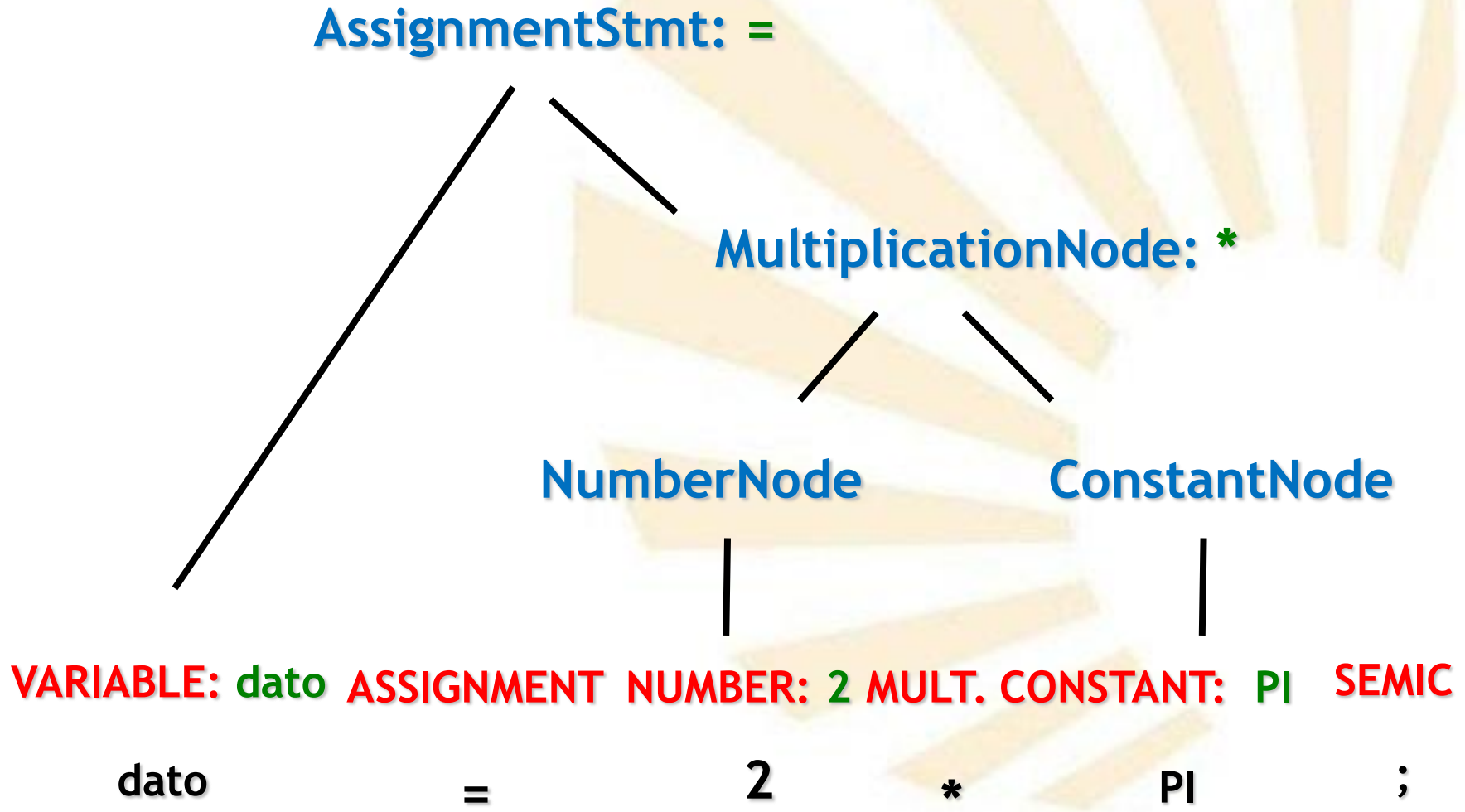
**dato**        =        **2**        \*        **PI**        ;

Ejemplo de la gramática 15: dato = 2 \* PI ;    Árbol sintáctico



# Ejemplo de la gramática 15: dato = 2 \* PI ;

AST de stmt



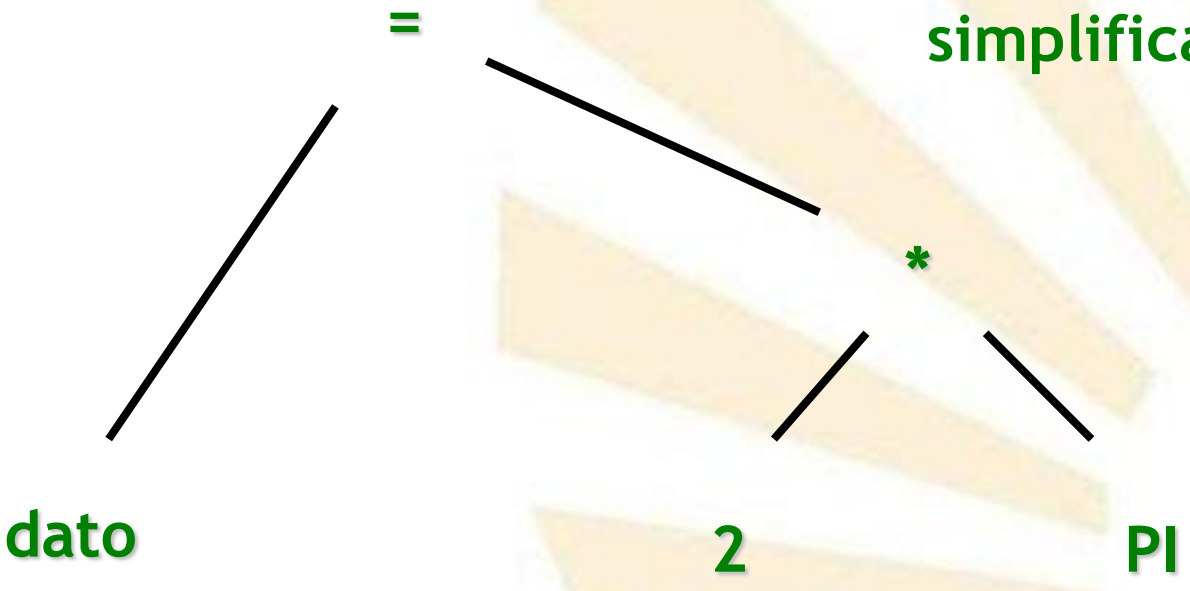
# Ejemplo de la gramática 15: dato = 2 \* PI ;

AST de stmt

Generación de AST

Árbol de sintaxis abstracta

simplificado



VARIABLE: dato ASSIGNMENT NUMBER: 2 MULT. CONSTANT: PI SEMIC

dato = 2 \* PI ;

## Ejemplo de la gramática 15: dato = 2 \* PI ;

### Derivación por la derecha en orden inverso

=> stmtlist

=> stmtlist stmt

=> stmtlist asgn SEMICOLON

=> stmtlist VARIABLE ASSIGNMENT exp SEMICOLON

=> stmtlist VARIABLE ASSIGNMENT exp MULTIPLICATION exp SEMICOLON

=> stmtlist VARIABLE ASSIGNMENT exp MULTIPLICATION CONSTANT SEMICOLON

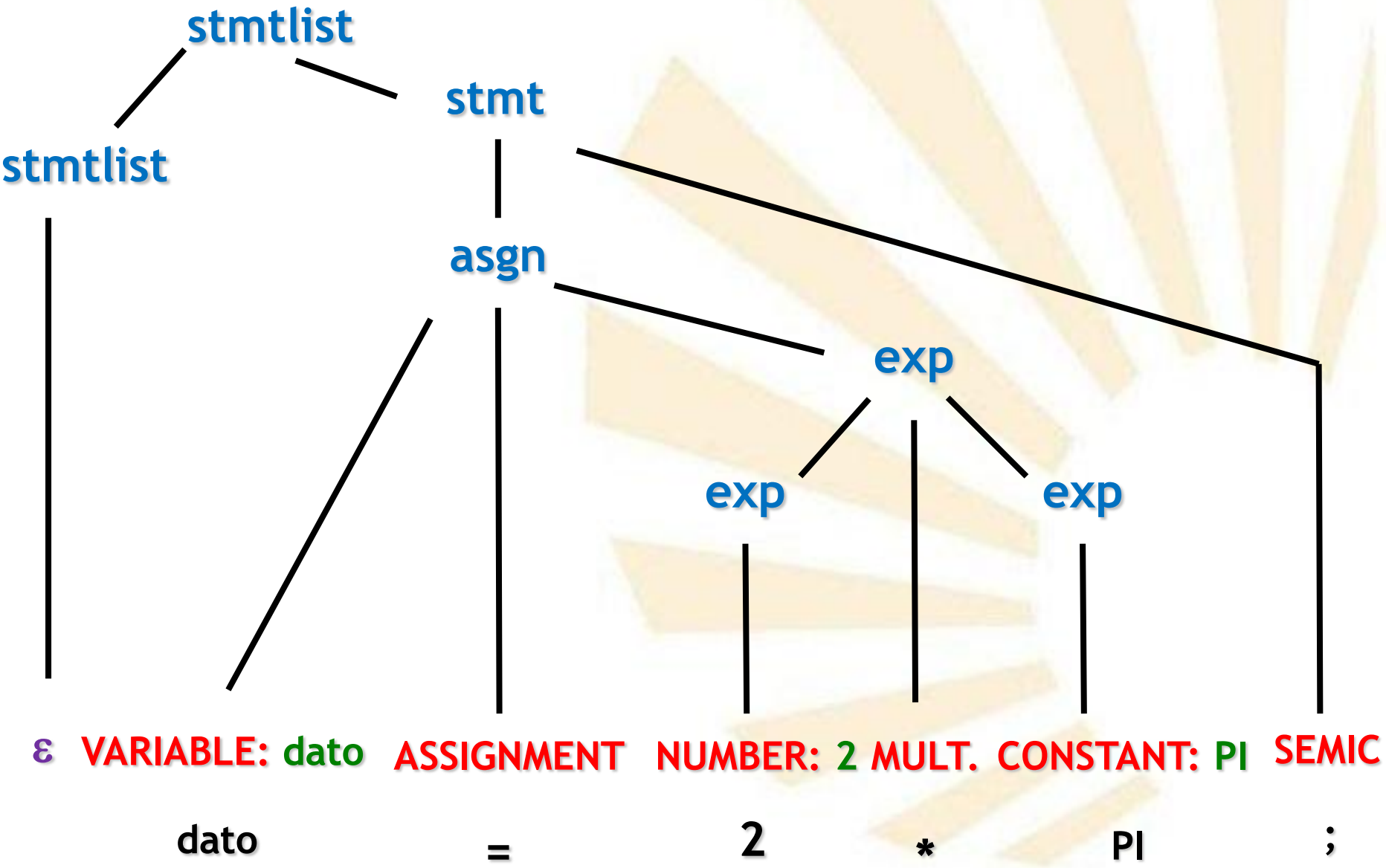
=> stmtlist VARIABLE ASSIGNMENT NUMBER MULTIPLICATION CONSTANT SEMICOLON

=> ε VARIABLE ASSIGNMENT NUMBER MULTIPLICATION CONSTANT SEMICOLON

= VARIABLE ASSIGNMENT NUMBER MULTIPLICATION CONSTANT SEMICOLON

dato = 2 \* PI ;

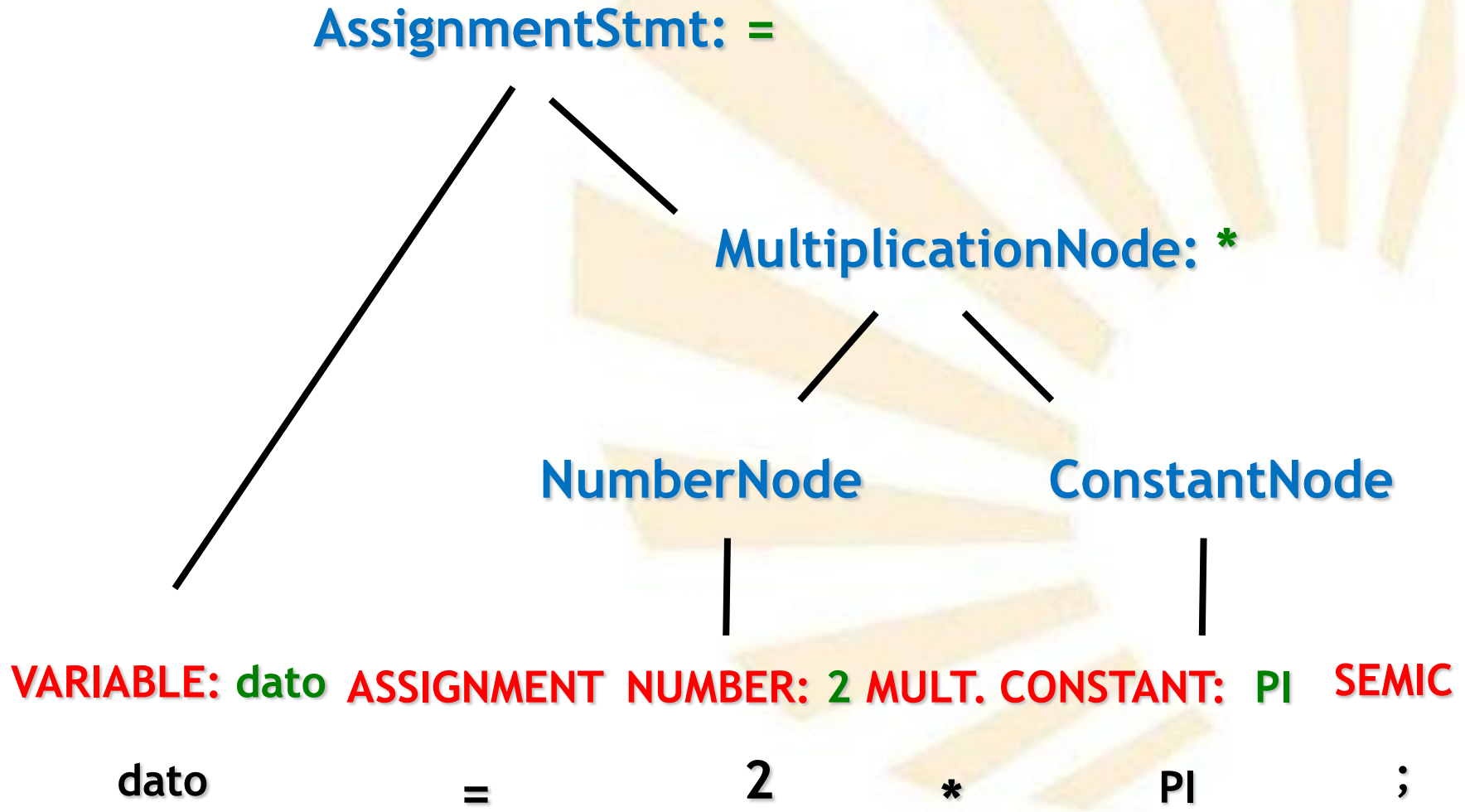
Ejemplo de la gramática 15: dato = 2 \* PI ;    Árbol sintáctico





# Ejemplo de la gramática 15: dato = 2 \* PI ;

## AST de stmtlist





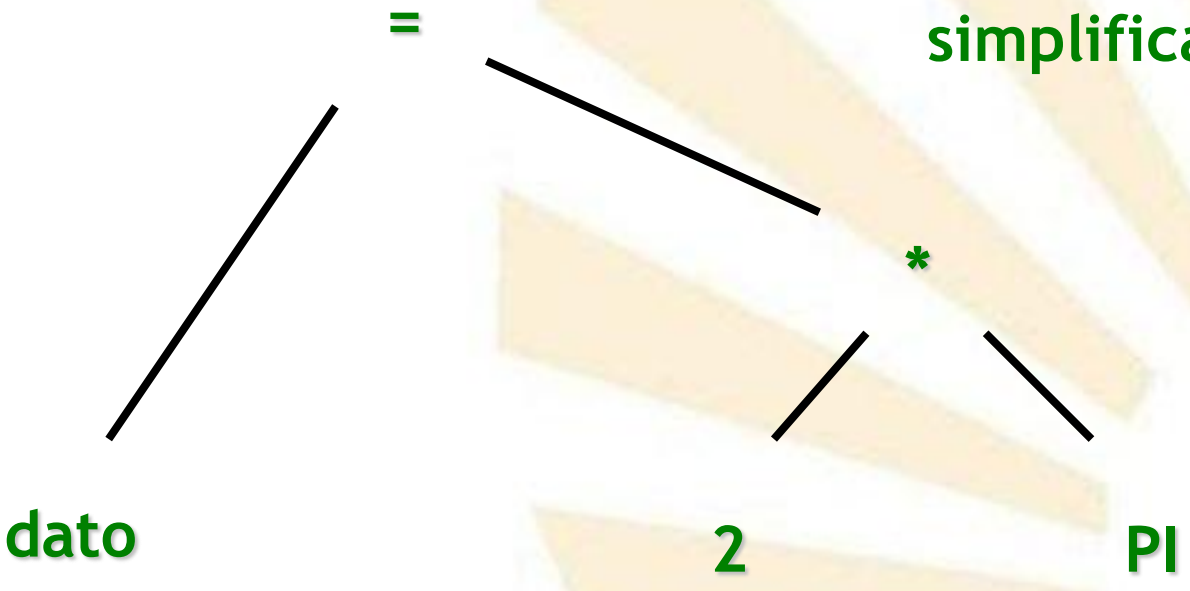
# Ejemplo de la gramática 15: dato = 2 \* PI ;

Generación de AST

Árbol de sintaxis abstracta

simplificado

AST de stmtlist



VARIABLE: dato ASSIGNMENT NUMBER: 2 MULT. CONSTANT: PI SEMIC

dato = 2 \* PI ;

# Ejemplo de la gramática 15: dato = 2 \* PI ;

## Derivación por la derecha en orden inverso

program

=> stmtlist

=> stmtlist stmt

=> stmtlist asgn SEMICOLON

=> stmtlist VARIABLE ASSIGNMENT exp SEMICOLON

=> stmtlist VARIABLE ASSIGNMENT exp MULTIPLICATION exp SEMICOLON

=> stmtlist VARIABLE ASSIGNMENT exp MULTIPLICATION CONSTANT SEMICOLON

=> stmtlist VARIABLE ASSIGNMENT NUMBER MULTIPLICATION CONSTANT SEMICOLON

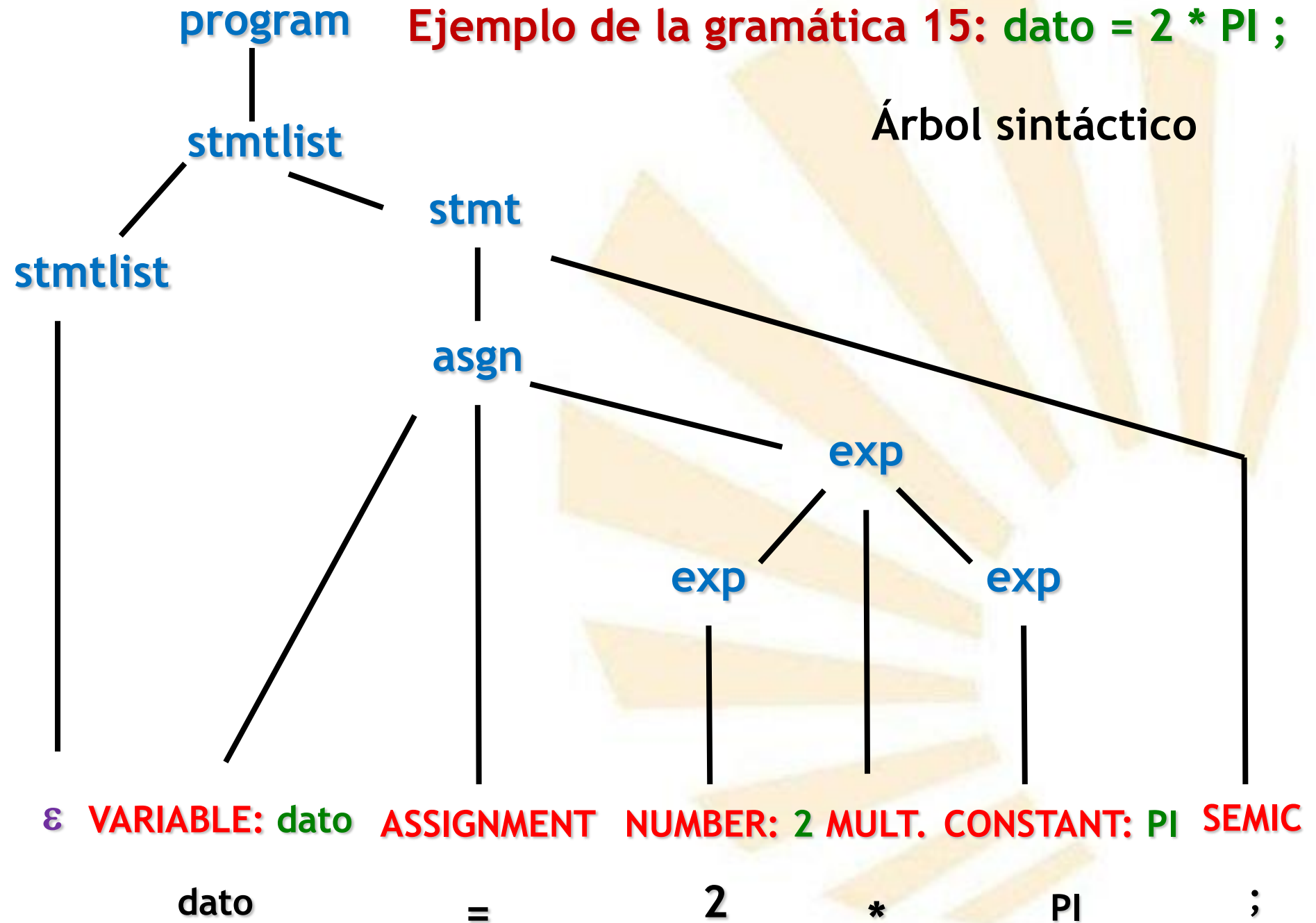
=> ε VARIABLE ASSIGNMENT NUMBER MULTIPLICATION CONSTANT SEMICOLON

= VARIABLE ASSIGNMENT NUMBER MULTIPLICATION CONSTANT SEMICOLON

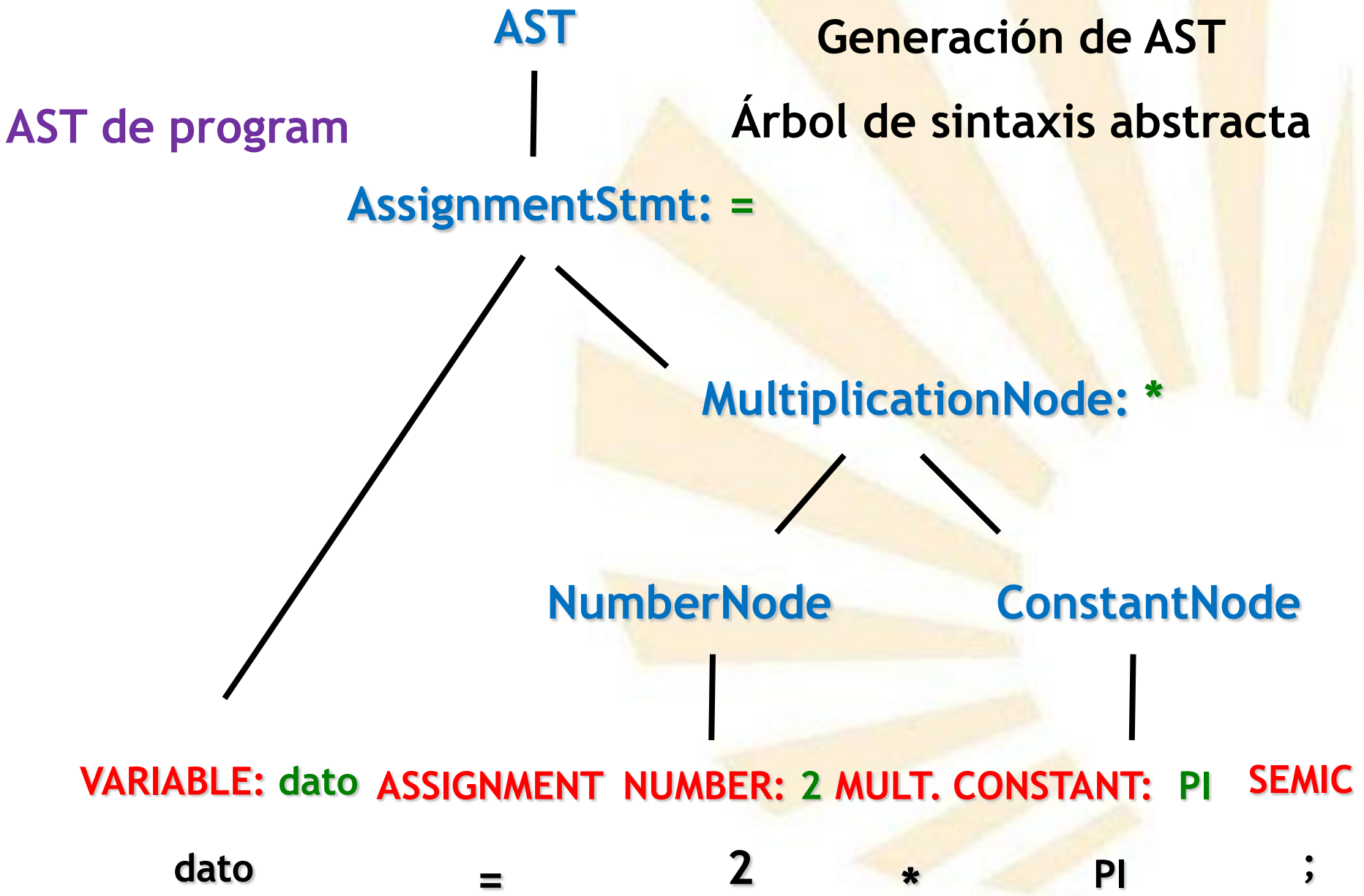
dato = 2 \* PI ;

Ejemplo de la gramática 15: dato = 2 \* PI ;

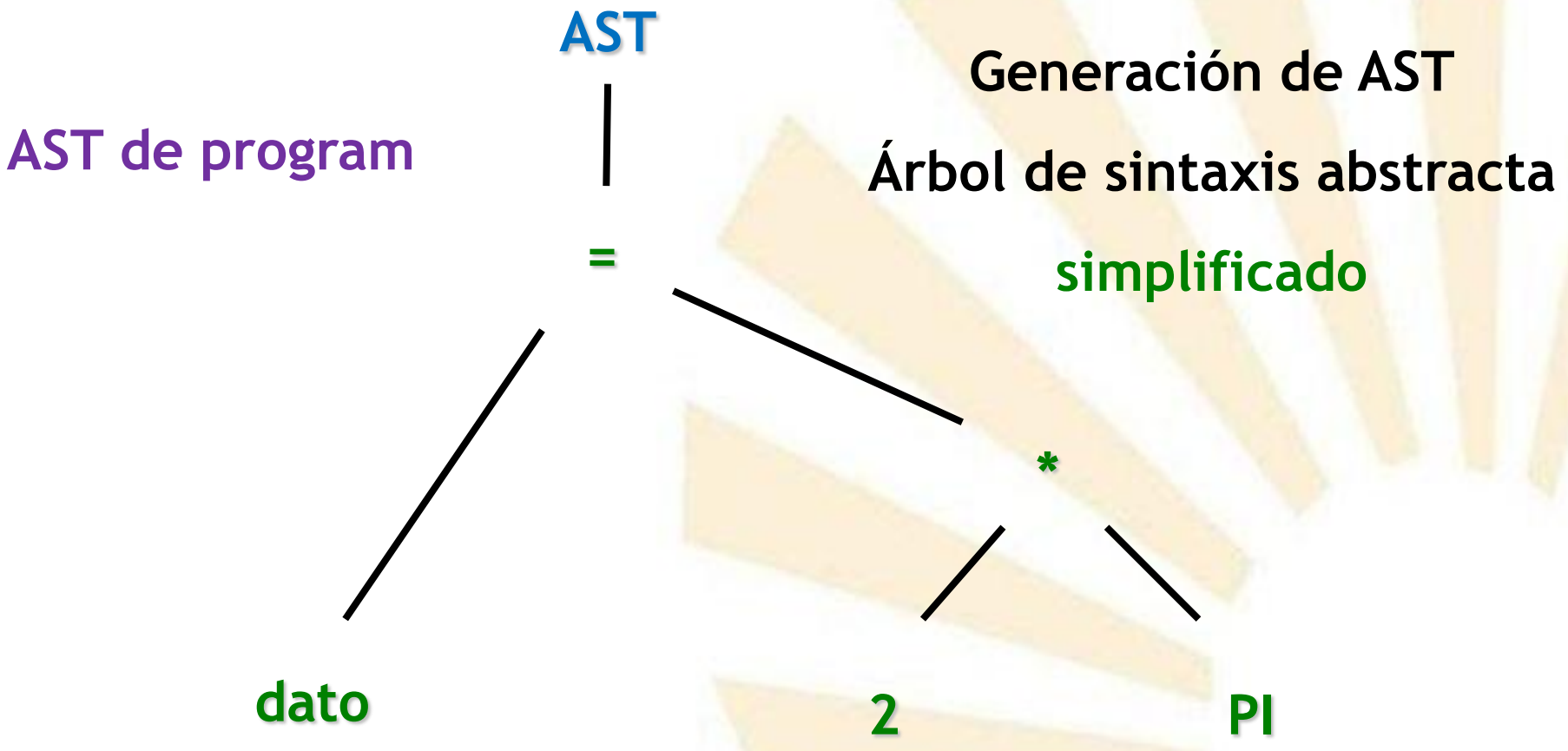
Árbol sintáctico



# Ejemplo de la gramática 15: dato = 2 \* PI ;



Ejemplo de la gramática 15: dato = 2 \* PI ;



VARIABLE: dato ASSIGNMENT NUMBER: 2 MULT. CONSTANT: PI SEMIC

dato = 2 \* PI ;

## Ejemplos

- Resumen inicial de ejemplos de AST
- Ejemplo de la gramática 15: `dato = 2 * PI;`
- **Ejemplo de la gramática 15: `print sin (PI / 2) ;`**
- Ejemplo de la gramática 16: `print (dato > 0) && (dato < 10) ;`
- Ejemplo de la gramática 17: `if (a>0) b = a; else b = -a;`
- Ejemplo de la gramática 17: `while (n>0) { print n; n = n-1;}`

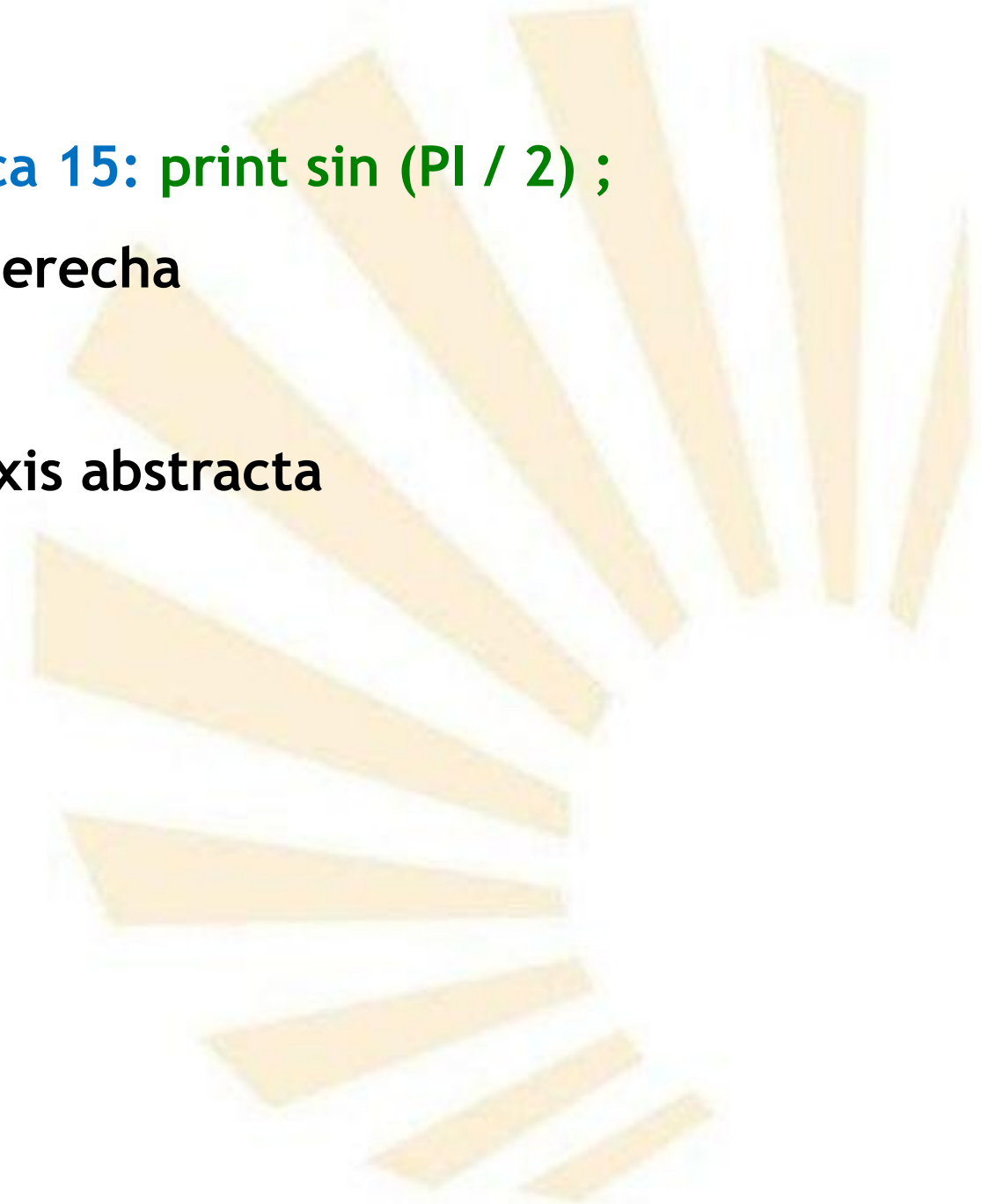
## Ejemplos

- Ejemplo de la gramática 15: `print sin (PI / 2) ;`



# Ejemplos

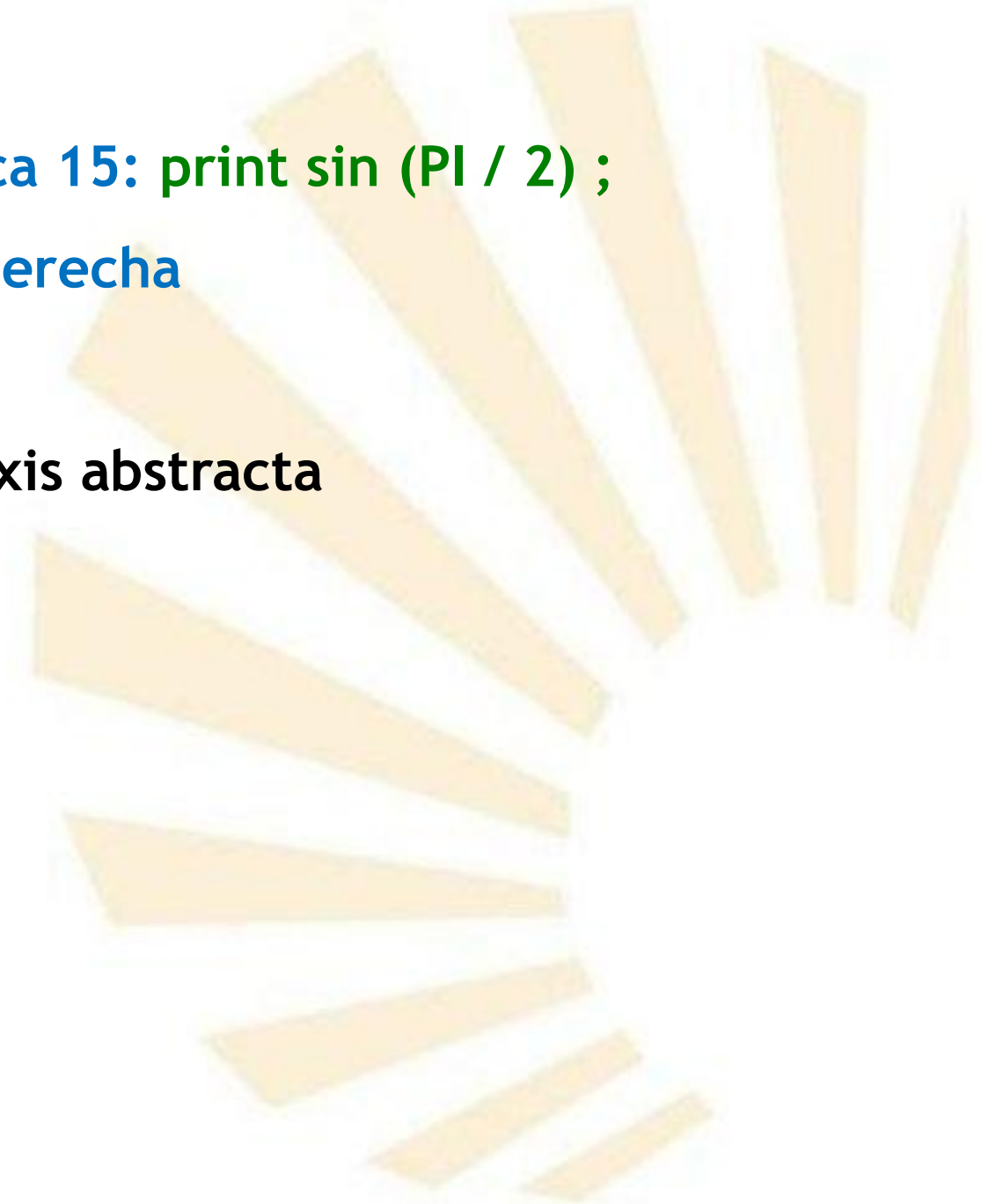
- **Ejemplo de la gramática 15: print sin (PI / 2) ;**
  - Derivación por la derecha
  - Árbol sintáctico
  - AST: árbol de sintaxis abstracta
  - AST simplificado





# Ejemplos

- Ejemplo de la gramática 15: **print sin (PI / 2) ;**
  - Derivación por la derecha
  - Árbol sintáctico
  - AST: árbol de sintaxis abstracta
  - AST simplificado



# Ejemplo de la gramática 15: `print sin(PI/2);`

## Derivación por la derecha

`program`

=> `stmtlist`

=> `stmtlist stmt`

=> `stmtlist print SEMICOLON`

=> `stmtlist PRINT exp SEMICOLON`

=> `stmtlist PRINT BUILTIN LPAREN listOfExp RPAREN SEMICOLON`

=> `stmtlist PRINT BUILTIN LPAREN exp restOfListOfExp RPAREN SEMICOLON`

=> `stmtlist PRINT BUILTIN LPAREN exp  $\epsilon$  RPAREN SEMICOLON`

=> `stmtlist PRINT BUILTIN LPAREN exp DIVISION exp RPAREN SEMICOLON`

=> `stmtlist PRINT BUILTIN LPAREN exp DIVISION NUMBER RPAREN SEMICOLON`

=> `stmtlist PRINT BUILTIN LPAREN CONSTANT DIVISION NUMBER RPAREN SEMICOLON`

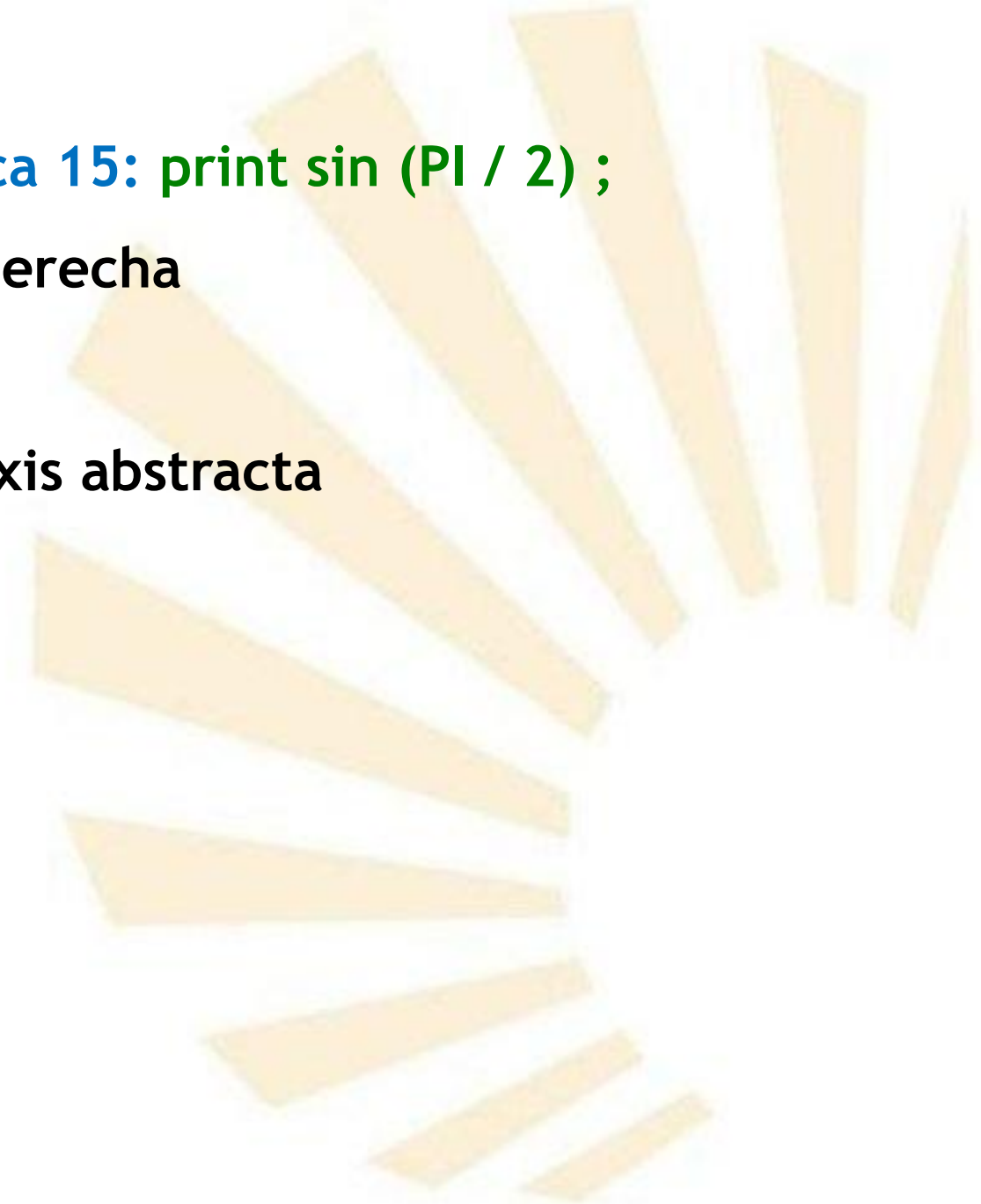
=  `$\epsilon$  PRINT BUILTIN LPAREN CONSTANT DIVISION NUMBER RPAREN SEMICOLON`

= `PRINT BUILTIN LPAREN CONSTANT DIVISION NUMBER RPAREN SEMICOLON`

`print sin ( PI / 2 ) ;`

# Ejemplos

- Ejemplo de la gramática 15: `print sin (PI / 2)` ;
  - Derivación por la derecha
  - **Árbol sintáctico**
  - **AST: árbol de sintaxis abstracta**
  - **AST simplificado**



# Ejemplo de la gramática 15:

```
print sin(PI/2);
```

program

Árbol sintáctico

stmtlist

stmt

print

exp

listOfExp

restOfListOfExp

exp

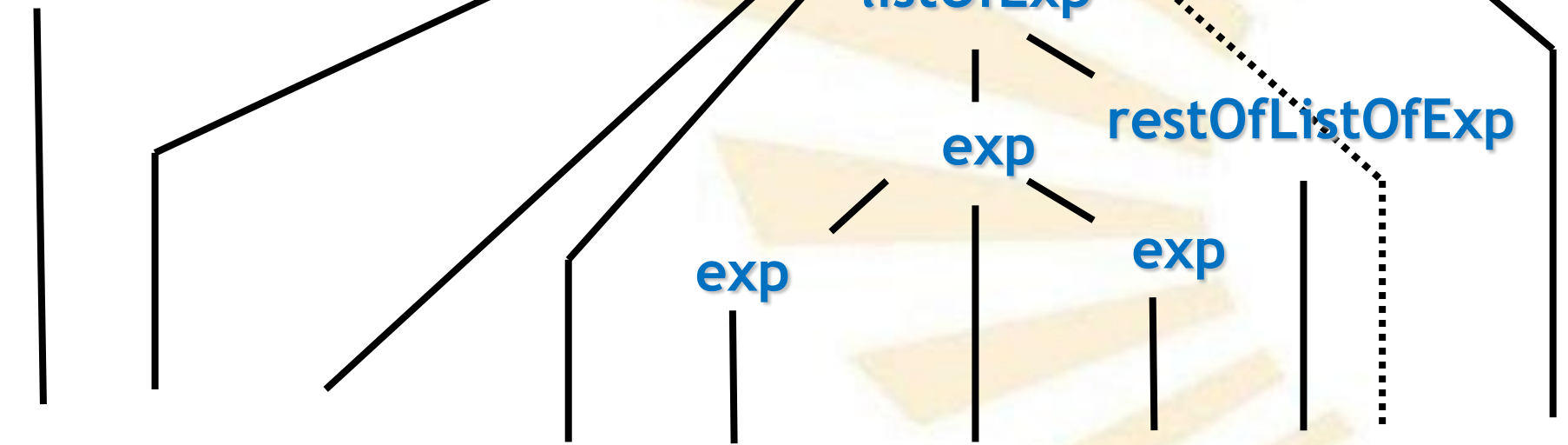
exp

exp

stmtlist

ε PRINT BUILTIN: sin LPAREN CONSTANT: PI DIVISION NUMBER: 2 ε RPAREN SEMIC

print sin ( PI / 2 ) ;



# Ejemplos

- Ejemplo de la gramática 15: `print sin (PI / 2)` ;
  - Derivación por la derecha
  - Árbol sintáctico
  - **AST: árbol de sintaxis abstracta**
  - **AST simplificado**

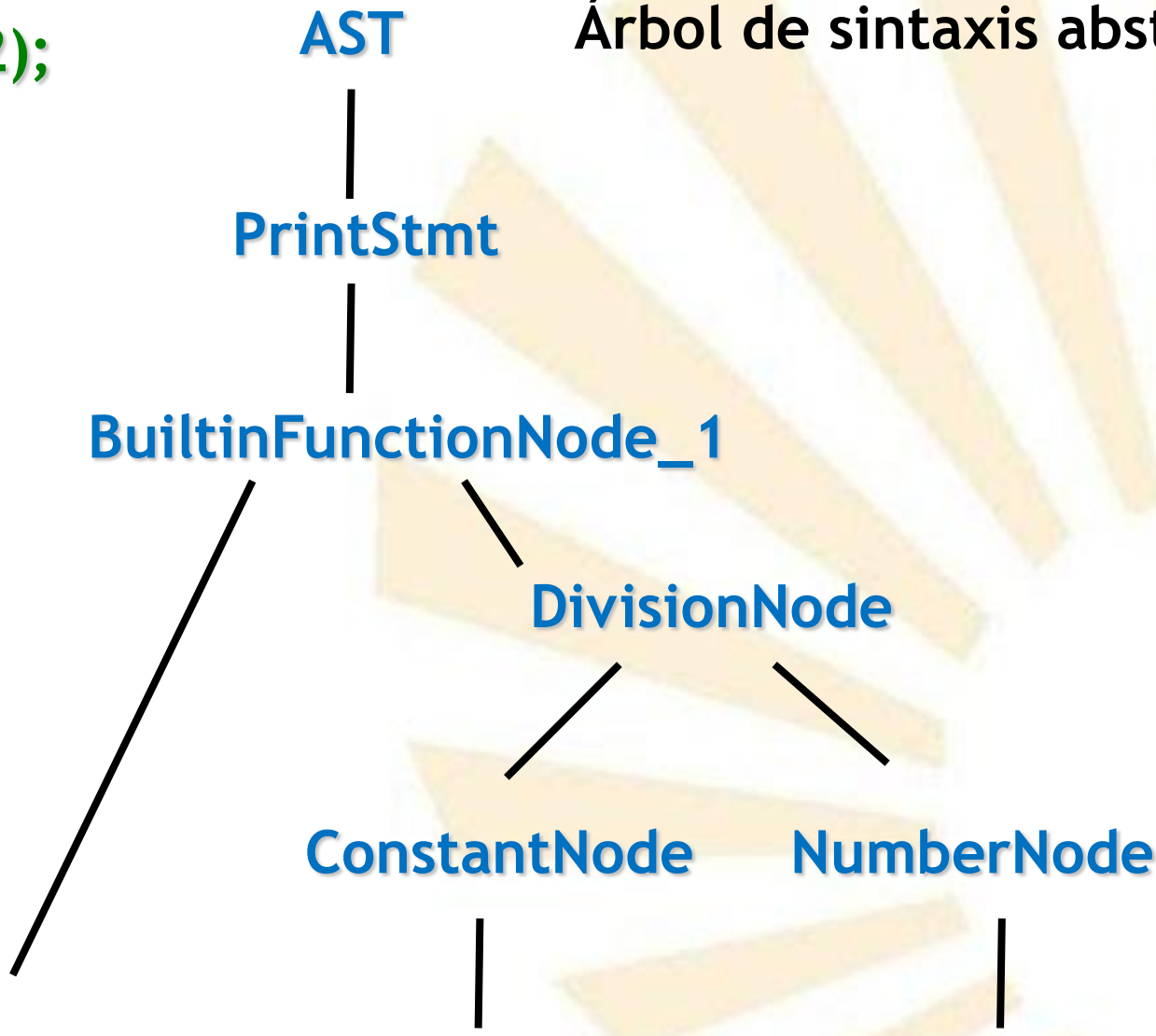


# Ejemplo de la gramática 15:

```
print sin(PI/2);
```

AST

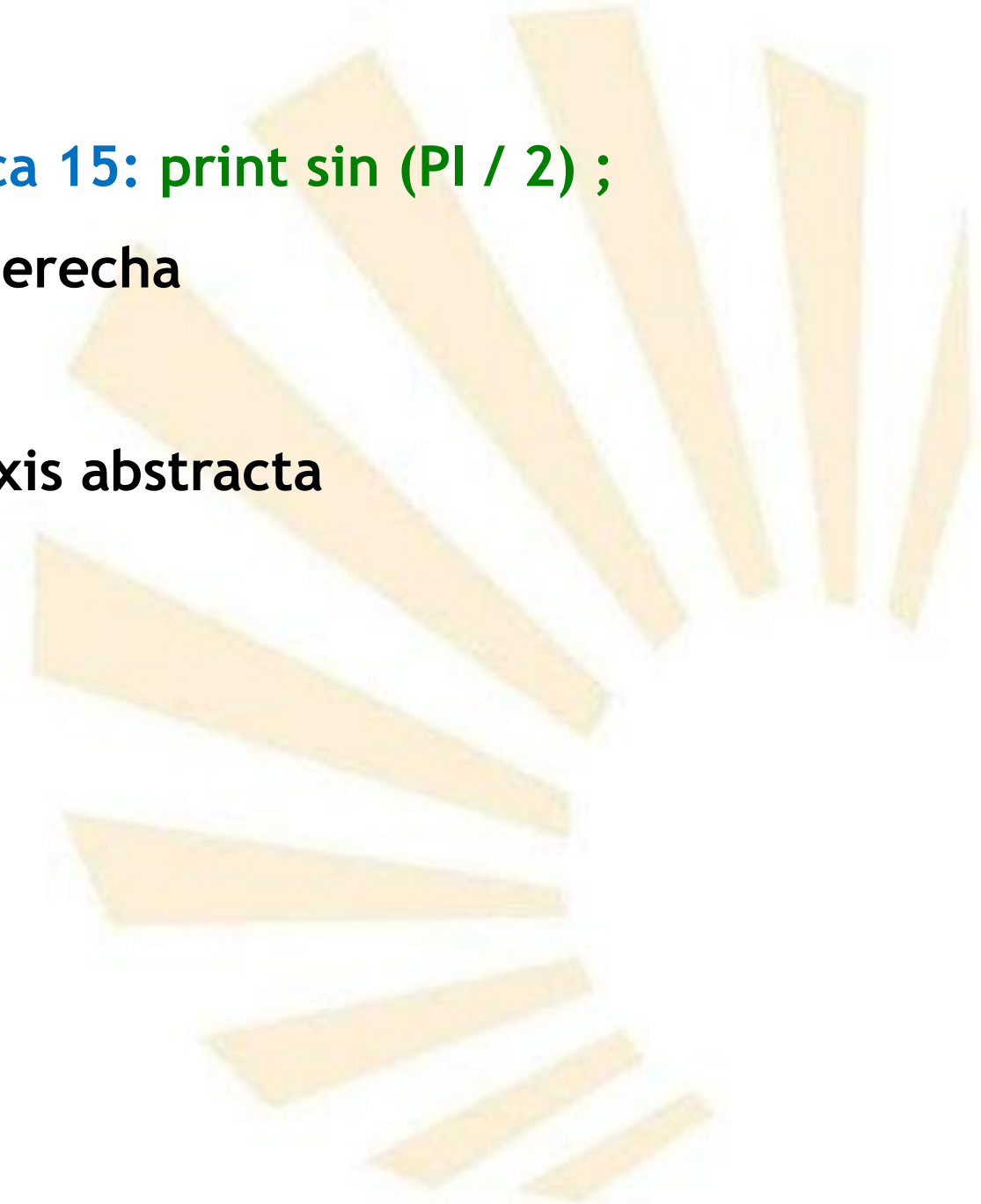
Árbol de sintaxis abstracta



PRINT BUILTIN: sin LPAREN CONSTANT: PI DIVISION NUMBER: 2 RPAREN SEMIC  
print sin ( PI / 2 ) ;

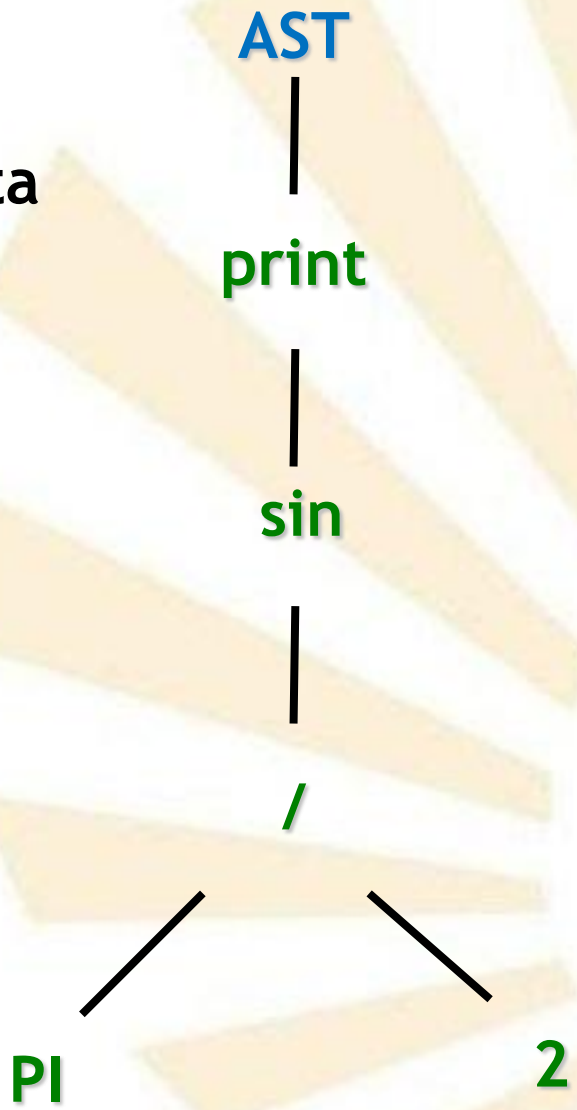
# Ejemplos

- **Ejemplo de la gramática 15: print sin (PI / 2) ;**
  - Derivación por la derecha
  - Árbol sintáctico
  - AST: árbol de sintaxis abstracta
  - **AST simplificado**



# Ejemplo de la gramática 15: print sin(PI/2);

AST  
Árbol de sintaxis abstracta  
simplificado



PRINT BUILTIN: sin LPAREN CONSTANT: PI DIVISION NUMBER: 2 RPAREN SEMIC  
print sin ( PI / 2 ) ;

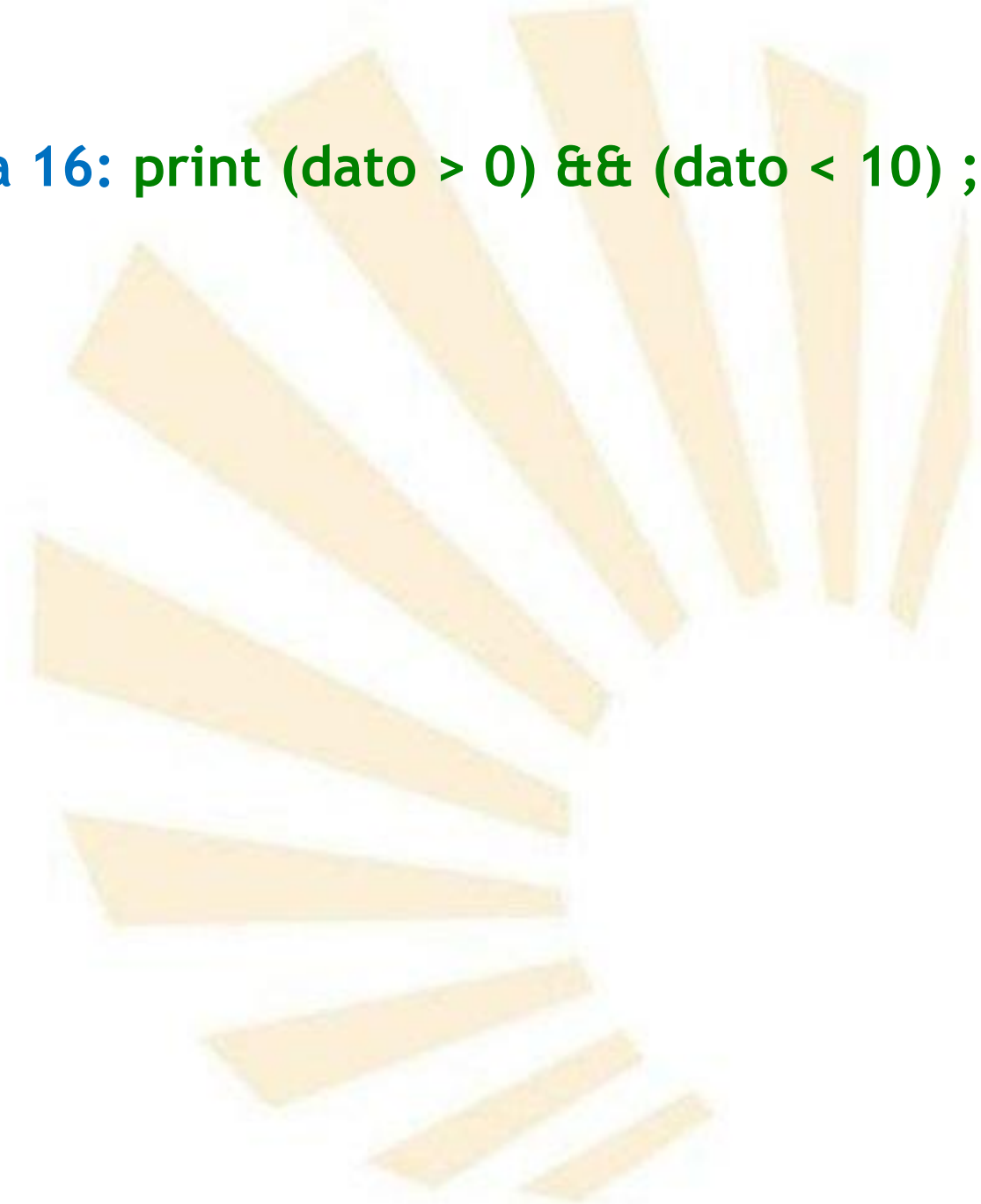


## Ejemplos

- Resumen inicial de ejemplos de AST
- Ejemplo de la gramática 15: `dato = 2 * PI;`
- Ejemplo de la gramática 15: `print sin (PI / 2) ;`
- **Ejemplo de la gramática 16: `print (dato > 0) && (dato < 10) ;`**
- Ejemplo de la gramática 17: `if (a>0) b = a; else b = -a;`
- Ejemplo de la gramática 17: `while (n>0) { print n; n = n-1;}`

## Ejemplos

- Ejemplo de la gramática 16: `print (dato > 0) && (dato < 10) ;`



## Ejemplos

- Ejemplo de la gramática 16: `print (dato > 0) && (dato < 10) ;`
  - Derivación por la derecha
  - Árbol sintáctico
  - AST: árbol de sintaxis abstracta
  - AST simplificado

# Ejemplos

- Ejemplo de la gramática 16: `print (dato > 0) && (dato < 10) ;`
  - Derivación por la derecha
  - Árbol sintáctico
  - AST: árbol de sintaxis abstracta
  - AST simplificado

# Ejemplo de la gramática 16: `print (dato > 0) && (dato < 10) ;`

## Derivación por la derecha

`program`

=> `stmtlist`

=> `stmtlist stmt`

=> `stmtlist print SEMICOLON`

=> `stmtlist PRINT exp SEMICOLON`

=> `stmtlist PRINT exp AND exp SEMICOLON`

=> `stmtlist PRINT exp AND LPAREN exp RPAREN SEMICOLON`

=> `stmtlist PRINT exp AND LPAREN exp L_T exp RPAREN SEMICOLON`

=> `stmtlist PRINT exp AND LPAREN exp L_T NUMBER RPAREN SEMICOLON`

=> `stmtlist PRINT exp AND LPAREN VAR L_T NUMBER RPAREN SEMICOLON`

=> `stmtlist PRINT LPAREN exp RPAREN AND LPAREN VAR L_T NUMBER RPAREN SEMICOLON`

=> `stmtlist PRINT LPAREN exp G_T exp RPAREN AND LPAREN VAR L_T NUMBER RPAREN SEMICOLON`

=> `stmtlist PRINT LPAREN exp G_T NUMBER RPAREN AND LPAREN VAR L_T NUMBER RPAREN SEMIC`

=> `stmtlist PRINT LPAREN VAR G_T NUMBER RPAREN AND LPAREN VAR L_T NUMBER RPAREN SEMIC`

=> `& PRINT LPAREN VAR G_T NUMBER RPAREN AND LPAREN VAR L_T NUMBER RPAREN SEMIC`

= `PRINT LPAREN VAR G_T NUMBER RPAREN AND LPAREN VAR L_T NUMBER RPAREN SEMIC`

`print ( dato > 0 ) && ( dato < 10 ) ;`

Abreviaturas:

`VAR = VARIABLE`

`G_T = GREATER_THAN`

`L_T = LESS_THAN`

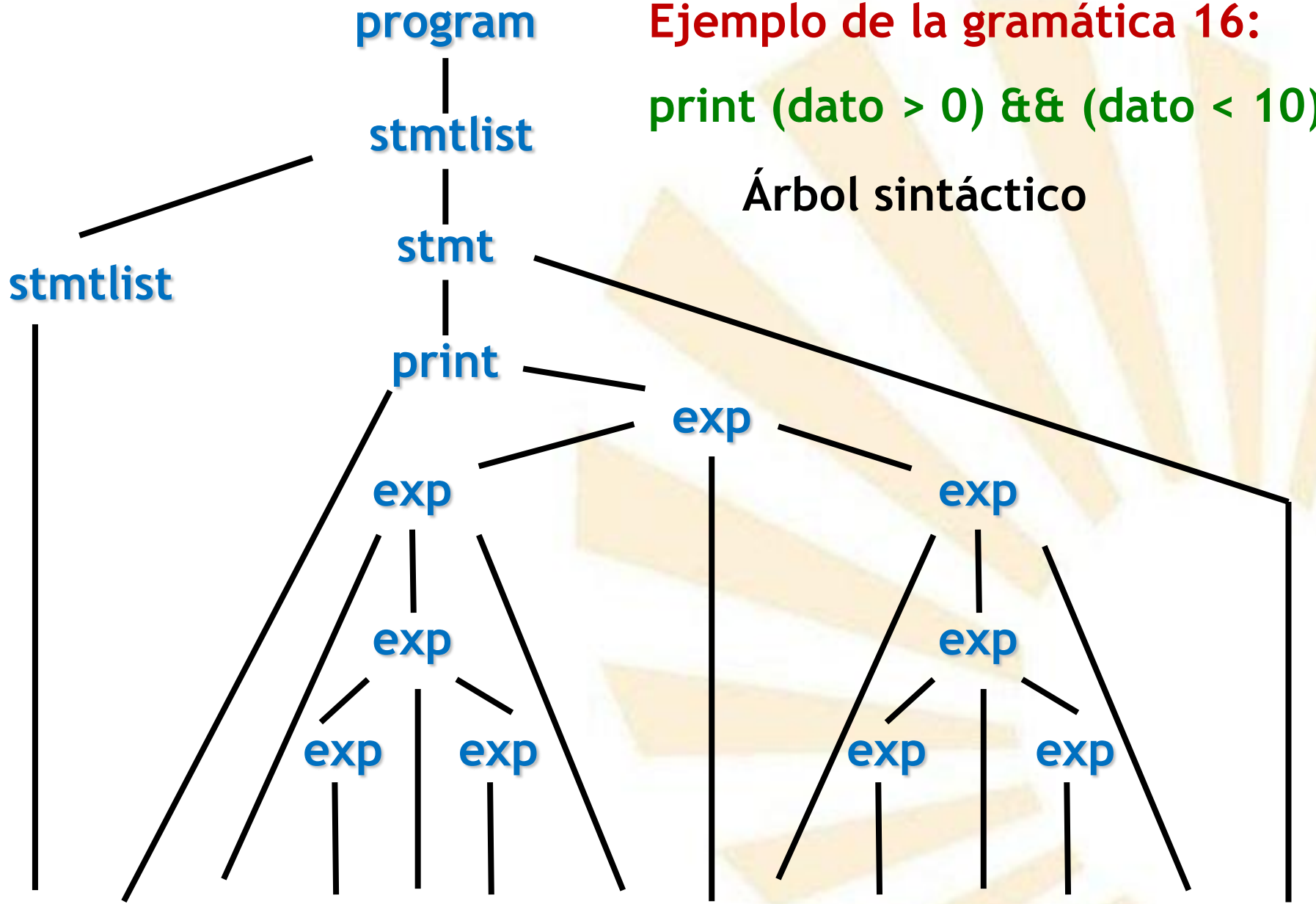
## Ejemplos

- Ejemplo de la gramática 16: `print (dato > 0) && (dato < 10) ;`
  - Derivación por la derecha
  - **Árbol sintáctico**
  - **AST: árbol de sintaxis abstracta**
  - **AST simplificado**

# Ejemplo de la gramática 16:

print (dato > 0) && (dato < 10) ;

Árbol sintáctico



ε PRINT LPAREN VAR G\_T NUMBER RPAREN AND LPAREN VAR L\_T NUMBER RPAREN SEMIC  
print ( dato > 0 ) && ( dato < 10 ) ;

## Ejemplos

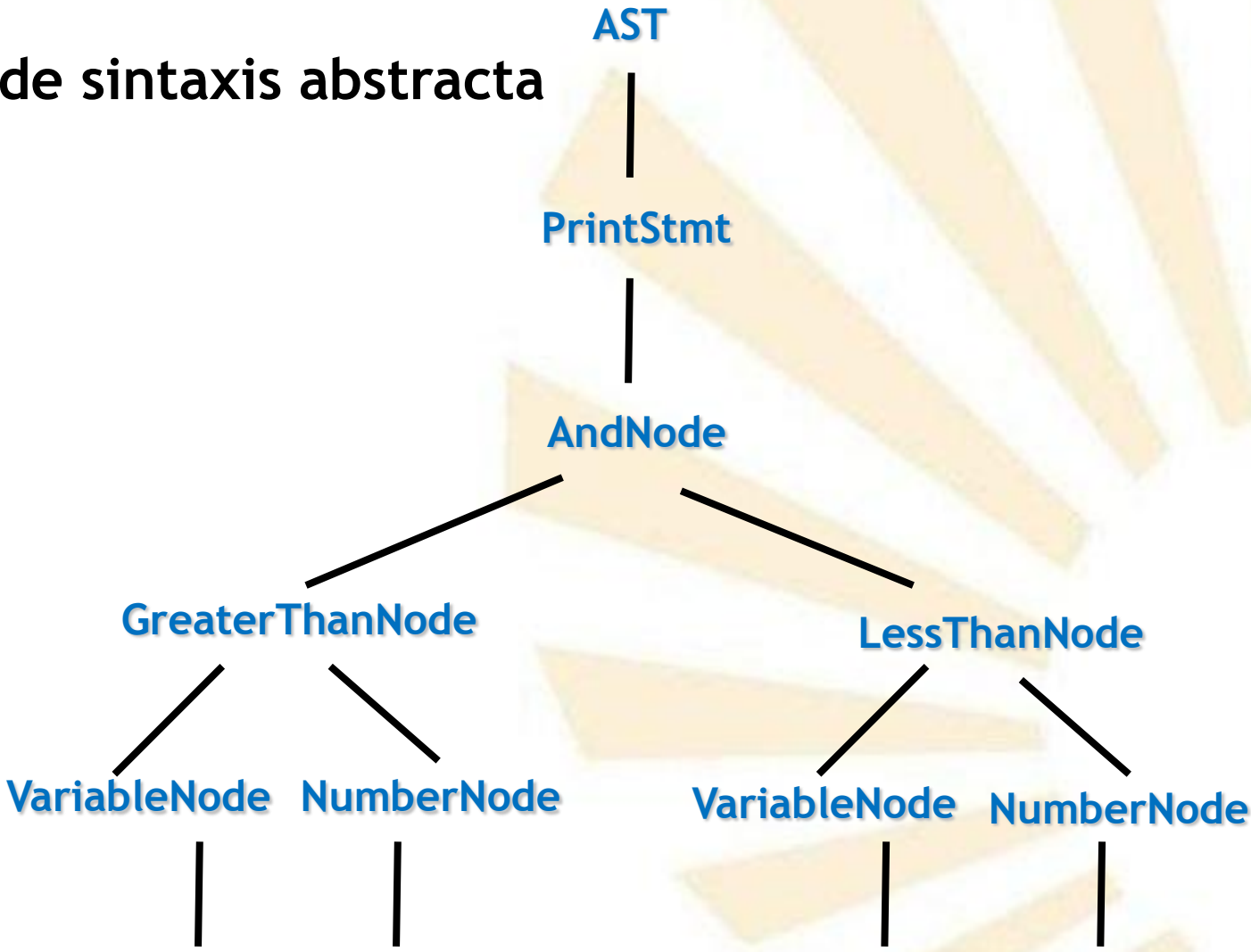
- Ejemplo de la gramática 16: `print (dato > 0) && (dato < 10) ;`
  - Derivación por la derecha
  - Árbol sintáctico
  - **AST: árbol de sintaxis abstracta**
  - **AST simplificado**



# Ejemplo de la gramática 16: `print (dato > 0) && (dato < 10) ;`

AST

Árbol de sintaxis abstracta



PRINT LPAREN VAR G\_T NUMBER RPAREN AND LPAREN VAR L\_T NUMBER RPAREN SEMIC  
print ( dato > 0 ) && ( dato < 10 ) ;

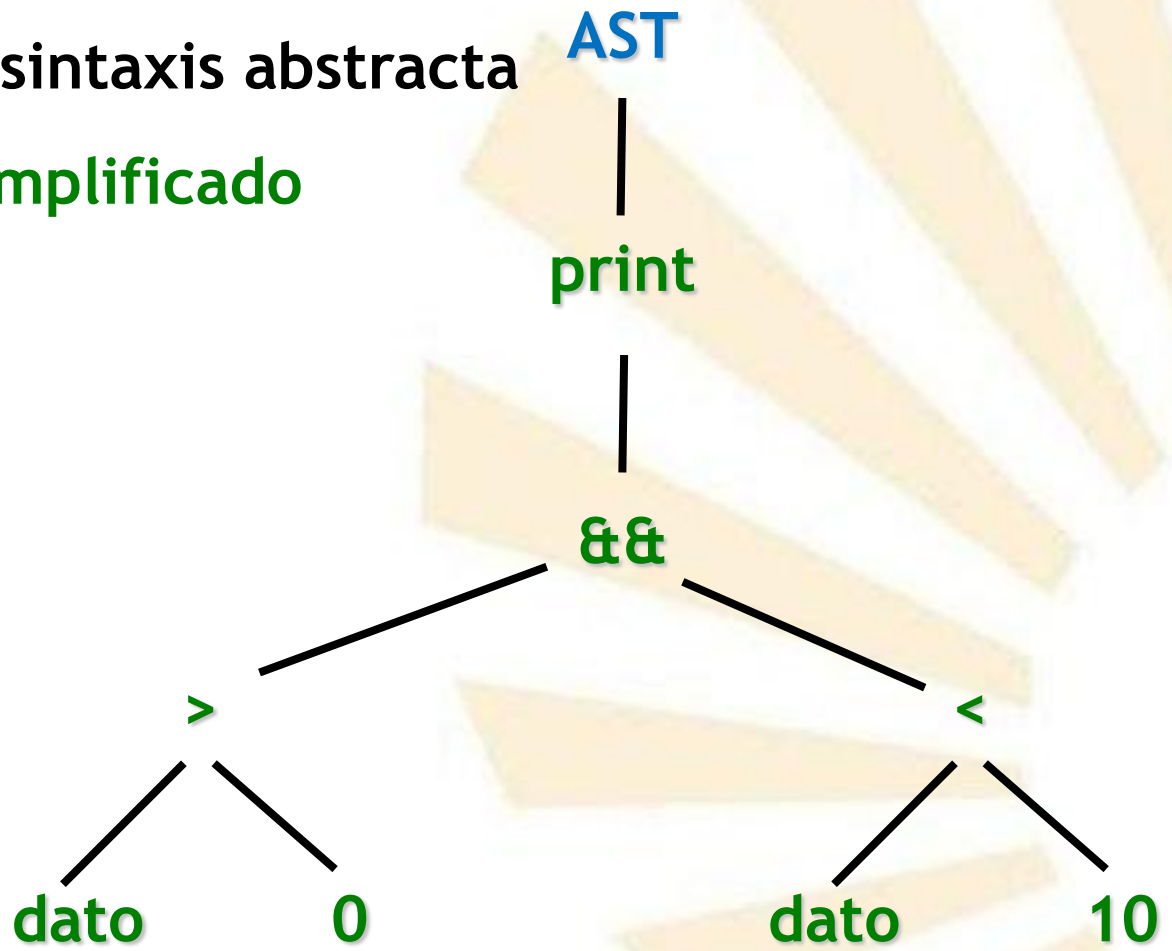
## Ejemplos

- Ejemplo de la gramática 16: `print (dato > 0) && (dato < 10) ;`
  - Derivación por la derecha en orden inverso
  - Árbol sintáctico
  - AST: árbol de sintaxis abstracta
  - AST simplificado

# Ejemplo de la gramática 16: `print (dato > 0) && (dato < 10) ;`

AST

Árbol de sintaxis abstracta  
simplificado



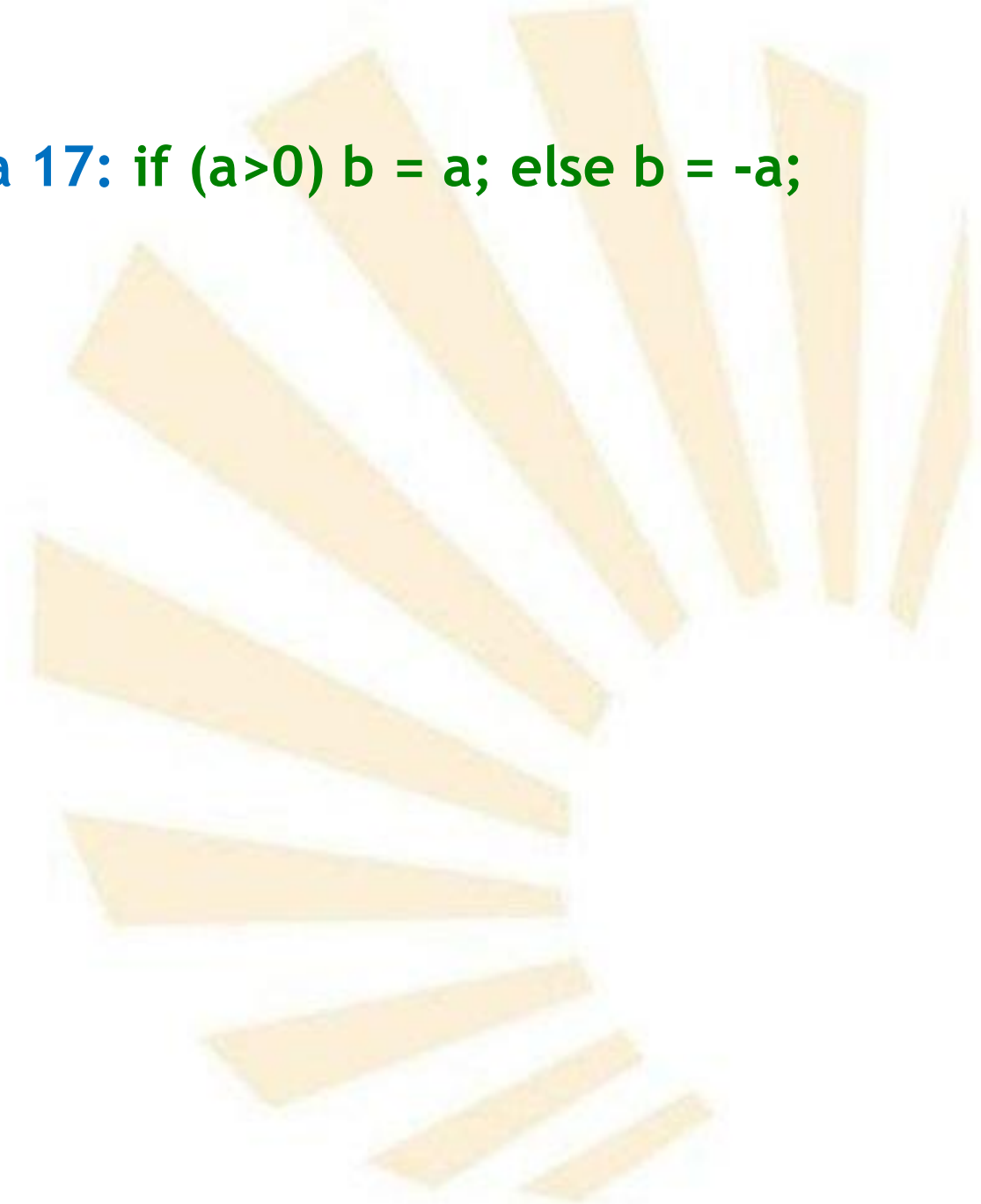
```
PRINT LPAREN VAR G_T NUMBER RPAREN AND LPAREN VAR L_T NUMBER RPAREN SEMIC  
print ( dato > 0 ) && ( dato < 10 ) ;
```

## Ejemplos

- Resumen inicial de ejemplos de AST
- Ejemplo de la gramática 15: `dato = 2 * PI;`
- Ejemplo de la gramática 15: `print sin (PI / 2) ;`
- Ejemplo de la gramática 16: `print (dato > 0) && (dato < 10) ;`
- **Ejemplo de la gramática 17: `if (a>0) b = a; else b = -a;`**
- Ejemplo de la gramática 17: `while (n>0) { print n; n = n-1;}`

## Ejemplos

- Ejemplo de la gramática 17: `if (a>0) b = a; else b = -a;`



## Ejemplos

- Ejemplo de la gramática 17: `if (a>0) b = a; else b = -a;`
  - Derivación por la derecha
  - Árbol sintáctico
  - AST: árbol de sintaxis abstracta
  - AST simplificado

## Ejemplos

- Ejemplo de la gramática 17: `if (a>0) b = a; else b = -a;`
  - Derivación por la derecha
  - Árbol sintáctico
  - AST: árbol de sintaxis abstracta
  - AST simplificado

## Ejemplos

- Ejemplo de la gramática 17: `if (a>0) b = a; else b = -a;`
  - Derivación por la derecha
    - Nota:
      - Se utiliza “controlSymbol” para controlar la ejecución interactiva de la sentencia “if”
      - Véase el código de `interprete.y`



# Ejemplo de la gramática 17: $\text{if } (a > 0) \text{ b} = \text{a}; \text{ else } \text{b} = -\text{a};$

program

## Derivación por la derecha

=> stmtlist

=> stmtlist stmt

=> stmtlist if

=> stmtlist IF cS cond stmt ELSE stmt

=> stmtlist IF cS cond stmt ELSE VAR ASSIG exp SEMICOLON

=> stmtlist IF cS cond stmt ELSE VAR ASSIG MINUS exp SEMICOLON

=> stmtlist IF cS cond stmt ELSE VAR ASSIG MINUS VAR SEMICOLON

=> stmtlist IF cS cond asgn SEMICOLON ELSE VAR ASSIG MINUS VAR SEMICOLON

=> stmtlist IF cS cond VAR ASSIG exp SEMICOLON ELSE VAR ASSIG MINUS VAR SEMICOLON

=> stmtlist IF cS cond VAR ASSIG VAR SEMICOLON ELSE VAR ASSIG MINUS VAR SEMICOLON

=> stmtlist IF cS LP exp RP VAR ASSIG VAR SEMICOLON ELSE VAR ASSIG MINUS VAR SEMICOLON

=> stmtlist IF cS LP exp G\_T exp RP VAR ASSIG VAR SEMIC ELSE VAR ASSIG MINUS VAR SEMIC

=> stmtlist IF cS LP exp G\_T N RP VAR ASSIG VAR SEMIC ELSE VAR ASSIG MINUS VAR SEMIC

=> stmtlist IF cS LP VAR G\_T N RP VAR ASSIG VAR SEMIC ELSE VAR ASSIG MINUS VAR SEMIC

=> stmtlist IF  $\epsilon$  LP VAR G\_T N RP VAR ASSIG VAR SEMIC ELSE VAR ASSIG MINUS VAR SEMIC

=  $\epsilon$  IF LP VAR G\_T N RP VAR ASSIG VAR SEMIC ELSE VAR ASSIG MINUS VAR SEMIC

if ( a > 0 ) b = a ; else b = - a ;

Abreviaturas:

cS = controlSymbol

VAR = VARIABLE

LP = LPAREN

RP = RPAREN

N = NUMBER

ASSIG = ASSIGNMENT

SEMIC = SEMICOLON

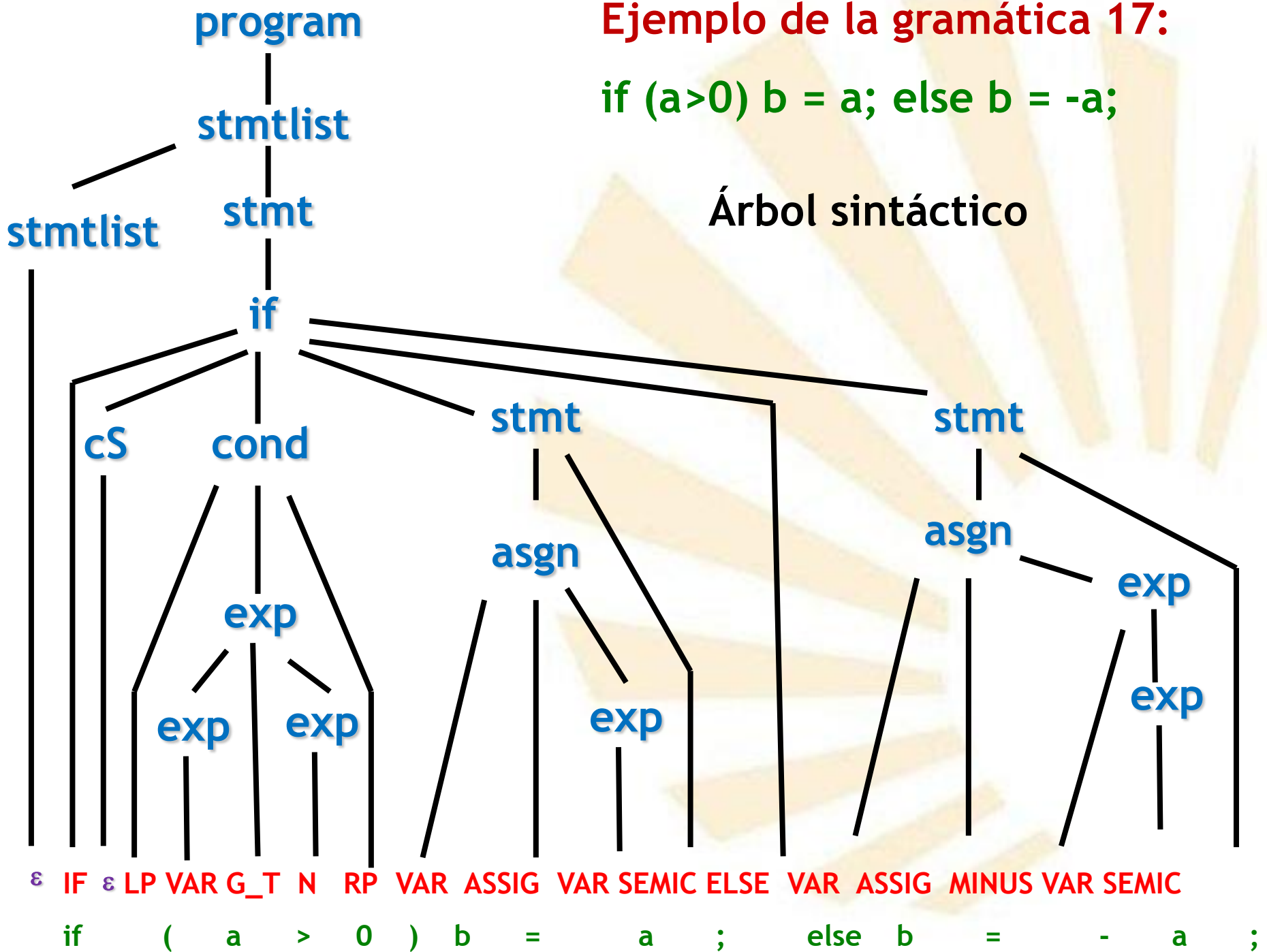
## Ejemplos

- Ejemplo de la gramática 17: `if (a>0) b = a; else b = a;`
  - Derivación por la derecha
  - **Árbol sintáctico**
  - **AST: árbol de sintaxis abstracta**
  - **AST simplificado**

# Ejemplo de la gramática 17:

if (a>0) b = a; else b = -a;

## Árbol sintáctico



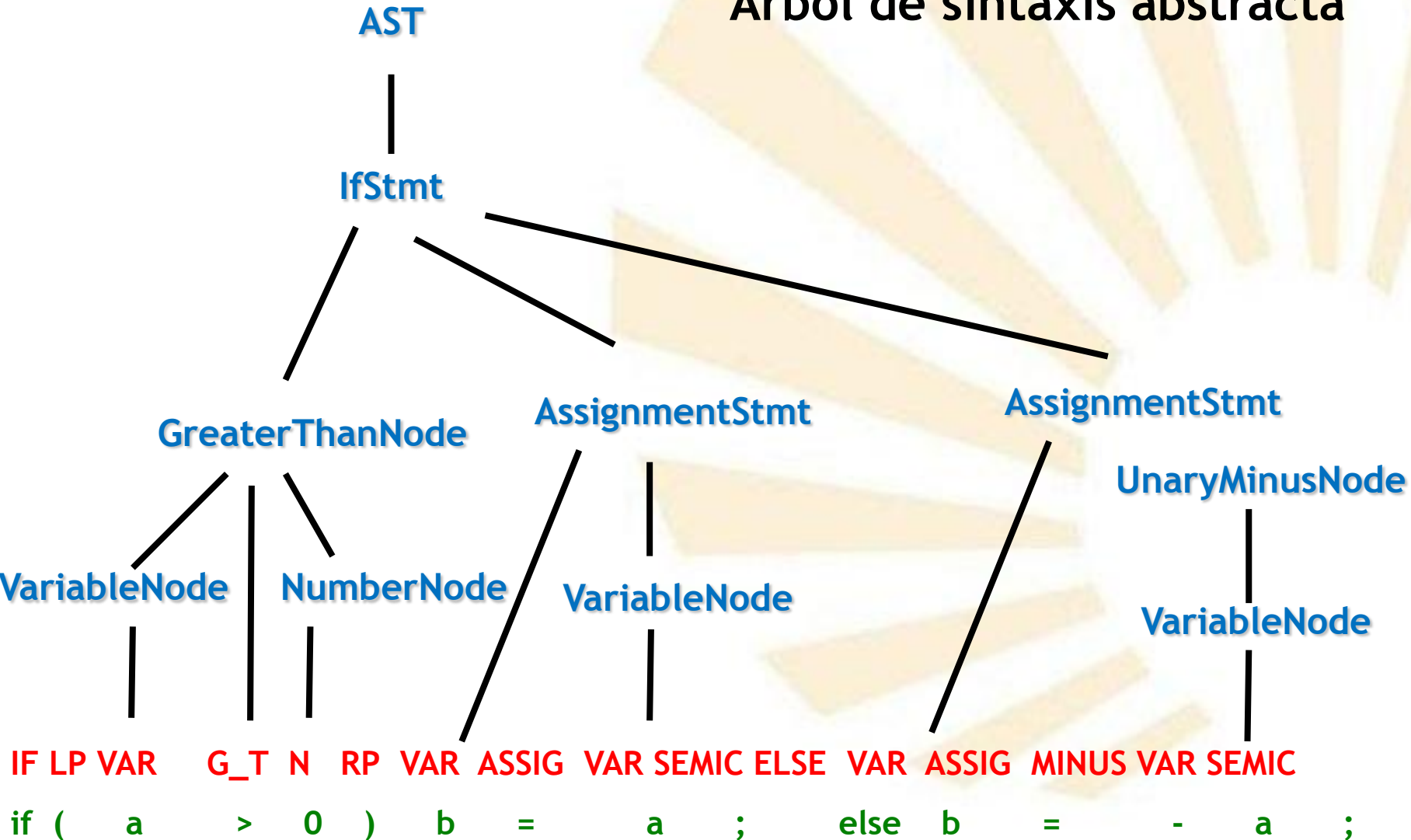
## Ejemplos

- Ejemplo de la gramática 17: `if (a>0) b = a; else b = -a;`
  - Derivación por la derecha
  - Árbol sintáctico
  - **AST: árbol de sintaxis abstracta**
  - **AST simplificado**

# Ejemplo de la gramática 17: `if (a>0) b = a; else b = -a;`

AST

Árbol de sintaxis abstracta



## Ejemplos

- Ejemplo de la gramática 17: `if (a>0) b = a; else b = -a;`
  - Derivación por la derecha en orden inverso
  - Árbol sintáctico
  - AST: árbol de sintaxis abstracta
  - AST simplificado

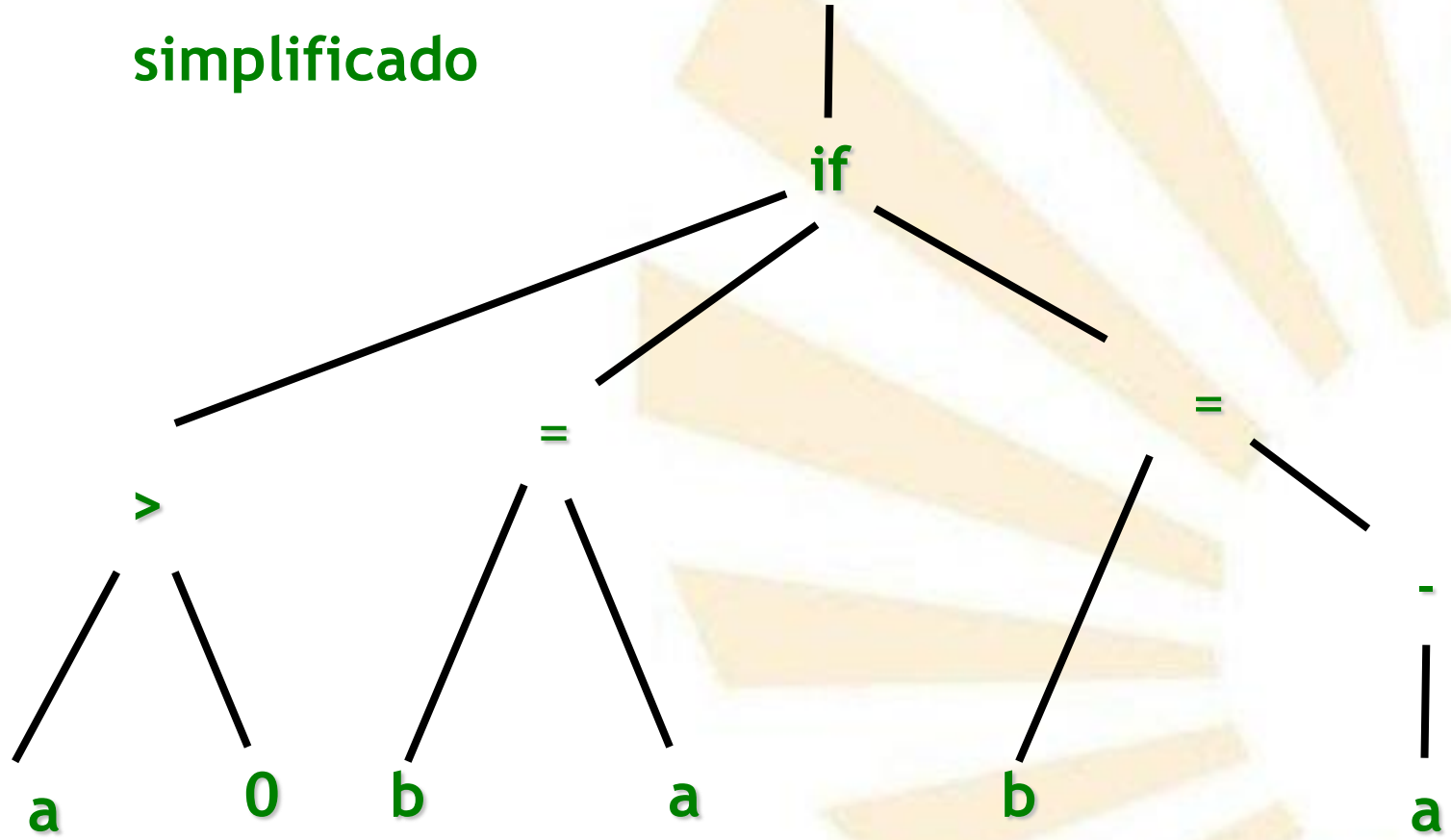
# Ejemplo de la gramática 17: `if (a>0) b = a; else b = -a;`

AST

Árbol de sintaxis abstracta

simplificado

AST



IF LP VAR G\_T N RP VAR ASSIG VAR SEMIC ELSE VAR ASSIG MINUS VAR SEMIC  
if ( a > 0 ) b = a ; else b = - a ;

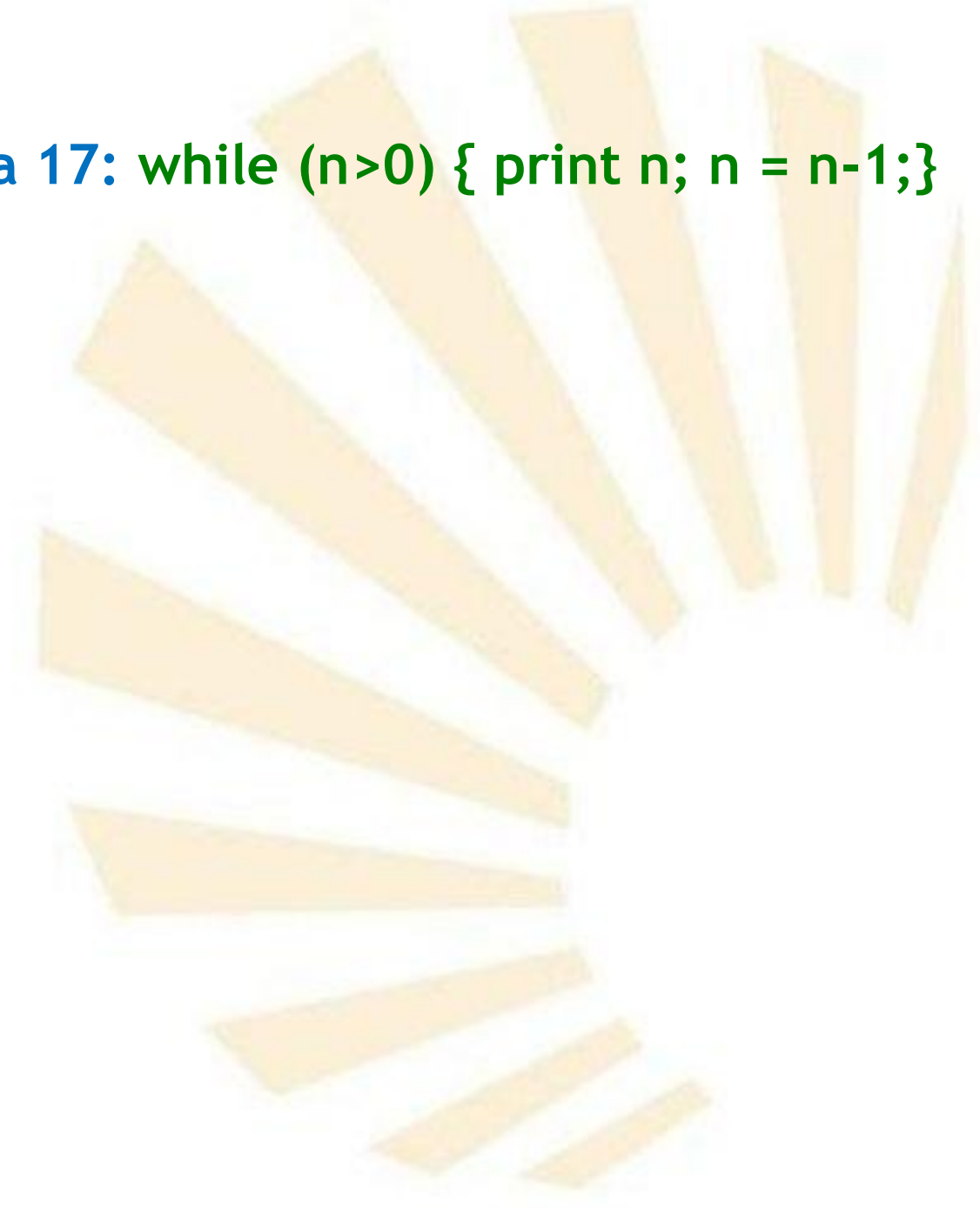
## Ejemplos

- Ejemplo de la gramática 15: `dato = 2 * PI;`
- Ejemplo de la gramática 15: `print sin (PI / 2);`
- Ejemplo de la gramática 16: `print (dato > 0) && (dato < 10);`
- Ejemplo de la gramática 17: `if (a>0) b = a; else b = -a;`
- Ejemplo de la gramática 17: `while (n>0) { print n; n = n-1;}`



## Ejemplos

- Ejemplo de la gramática 17: `while (n>0) { print n; n = n-1;}`



## Ejemplos

- Ejemplo de la gramática 17: `while (n>0) { print n; n = n-1;}`
  - Derivación por la derecha
  - Árbol sintáctico
  - AST: árbol de sintaxis abstracta
  - AST simplificado

## Ejemplos

- Ejemplo de la gramática 17: `while (n>0) { print n; n = n-1;}`
  - Derivación por la derecha
  - Árbol sintáctico
  - AST: árbol de sintaxis abstracta
  - AST simplificado

## Ejemplos

- Ejemplo de la gramática 17: `while (n>0) { print n; n = n-1;}`
  - Derivación por la derecha
    - Nota:
      - Se utiliza “controlSymbol” para controlar la ejecución interactiva de la sentencia “while”
      - Véase el código de interprete.y

# Ejemplo de la gramática 17: `while (n>0) { print n; n = n-1;}`

program

## Derivación por la derecha

=> stmtlist

=> stmtlist stmt

=> stmtlist while

=> stmtlist WHILE cS cond stmt

=> stmtlist WHILE cS cond block

=> stmtlist WHILE cS cond LB stmtlist RB

=> stmtlist WHILE cS cond LB stmtlist stmt RB

=> stmtlist WHILE cS cond LB stmtlist asgn RB

=> stmtlist WHILE cS cond LB stmtlist VAR ASSIG exp SEMIC RB

=> stmtlist WHILE cS cond LB stmtlist VAR ASSIG exp MINUS exp SEMIC RB

=> stmtlist WHILE cS cond LB stmtlist VAR ASSIG exp MINUS N SEMIC RB

=> stmtlist WHILE cS cond LB stmtlist VAR ASSIG VAR MINUS N SEMIC RB

=> stmtlist WHILE cS cond LB stmtlist stmt VAR ASSIG VAR MINUS N SEMIC RB (continúa)

...

Abreviaturas:

cS = controlSymbol

VAR = VARIABLE

LP = LPAREN

RP = RPAREN

N = NUMBER

ASSIG = ASSIGNMENT

SEMIC = SEMICOLON

LEFTCURLYBRACKET = LB

RIGHTCURLYBRACKET = RB

# Ejemplo de la gramática 17: `while (n>0) { print n; n = n-1;}`

## Derivación por la derecha

=> stmtlist WHILE cS cond LB stmtlist **stmt** VAR ASSIG VAR MINUS N SEMIC RB (continúa)

=> stmtlist WHILE cS cond LB stmtlist **print** SEMIC VAR ASSIG VAR MINUS N SEMIC RB

=> stmtlist WHILE cS cond LB stmtlist PRINT **exp** SEMIC VAR ASSIG VAR MINUS N SEMIC RB

=> stmtlist WHILE cS cond LB **stmtlist** PRINT VAR SEMIC VAR ASSIG VAR MINUS N SEMIC RB

=> stmtlist WHILE cS **cond** LB  $\epsilon$  PRINT VAR SEMIC VAR ASSIG VAR MINUS N SEMIC RB

=> stmtlist WHILE cS LP **exp** RP LB PRINT VAR SEMIC VAR ASSIG VAR MINUS N SEMIC RB

=> stmtlist WHILE cS LP exp G\_T **exp** RP LB PRINT VAR SEMIC VAR ASSIG VAR MINUS N SEMIC RB

=> stmtlist WHILE cS LP **exp** G\_T N RP LB PRINT VAR SEMIC VAR ASSIG VAR MINUS N SEMIC RB

=> stmtlist WHILE **cS** LP VAR G\_T N RP LB PRINT VAR SEMIC VAR ASSIG VAR MINUS N SEMIC RB

=> **stmtlist** WHILE  $\epsilon$  LP VAR G\_T N RP LB PRINT VAR SEMIC VAR ASSIG VAR MINUS N SEMIC RB

=>  $\epsilon$  WHILE LP VAR G\_T N RP LB PRINT VAR SEMIC VAR ASSIG VAR MINUS N SEMIC RB

= **WHILE LP VAR G\_T N RP LB PRINT VAR SEMIC VAR ASSIG VAR MINUS N SEMIC RB**

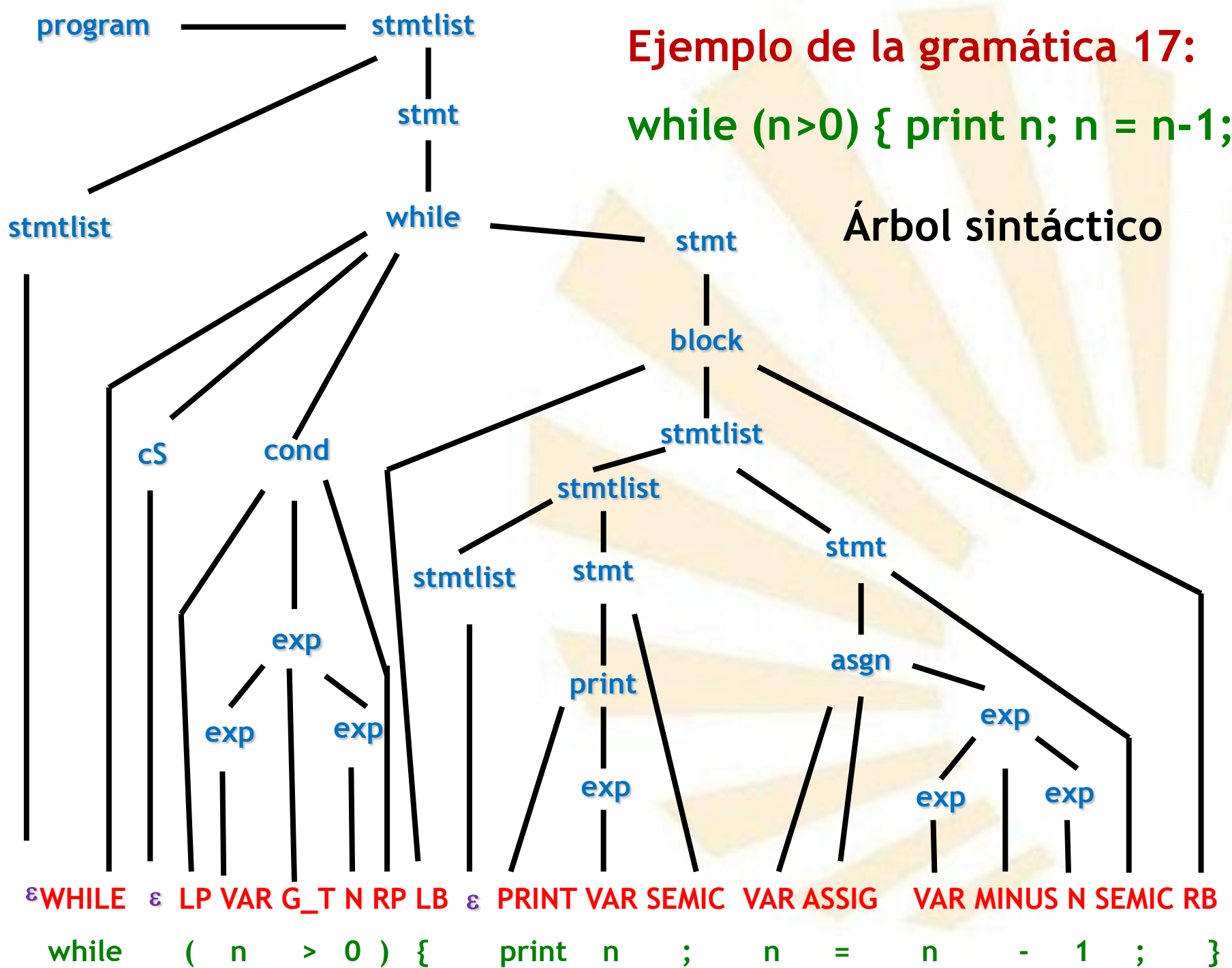
**while ( n > 0 ) { print n ; n = n - 1 ; }**

## Ejemplos

- Ejemplo de la gramática 17: `while (n>0) { print n; n = n-1;}`
  - Derivación por la derecha
  - **Árbol sintáctico**
  - **AST: árbol de sintaxis abstracta**
  - **AST simplificado**

**Ejemplo de la gramática 17:**  
**while (n>0) { print n; n = n-1;}**

**Árbol sintáctico**





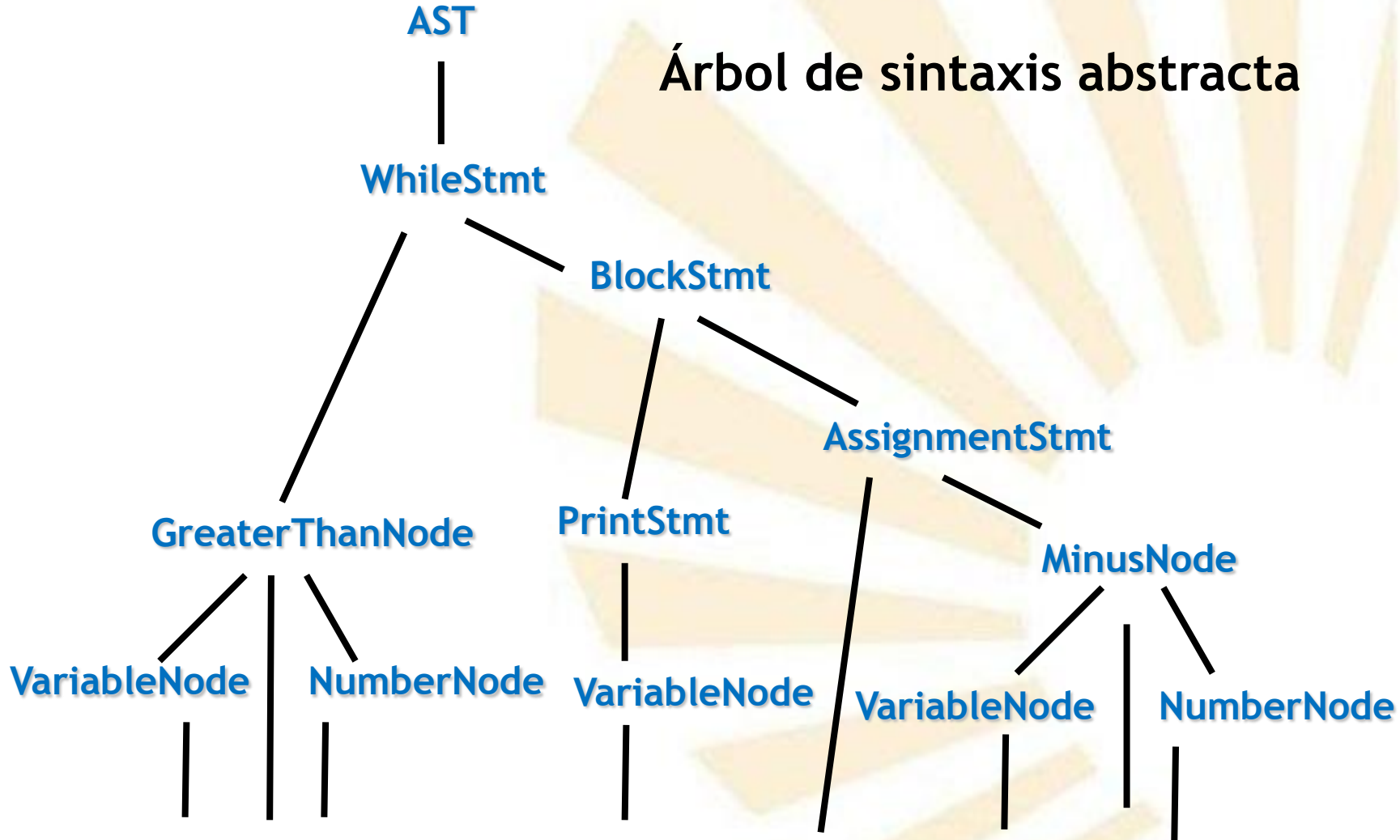
## Ejemplos

- Ejemplo de la gramática 17: `while (n>0) { print n; n = n-1;}`
  - Derivación por la derecha
  - Árbol sintáctico
  - **AST: árbol de sintaxis abstracta**
  - **AST simplificado**

# Ejemplo de la gramática 17: while (n>0) { print n; n = n-1;}

AST

Árbol de sintaxis abstracta



WHILE LP VAR G\_T N RP LB & PRINT VAR SEMIC VAR ASSIG VAR MINUS N SEMIC RB

while ( n > 0 ) { print n ; n = n - 1 ; }

## Ejemplos

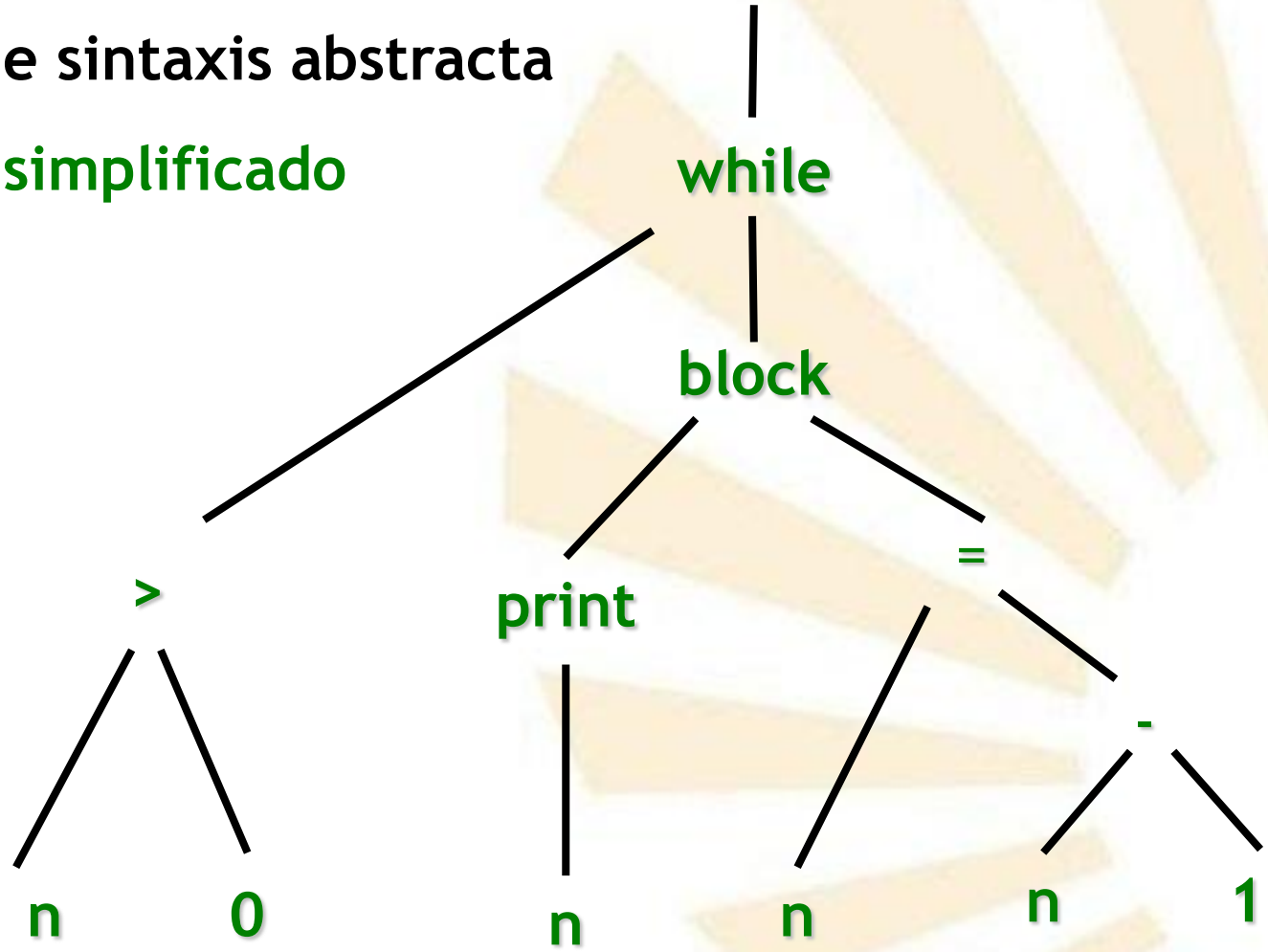
- Ejemplo de la gramática 17: `while (n>0) { print n; n = n-1;}`
  - Derivación por la derecha en orden inverso
  - Árbol sintáctico
  - AST: árbol de sintaxis abstracta
  - AST simplificado

# Ejemplo de la gramática 17: while (n>0) { print n; n = n-1; }

AST

AST

Árbol de sintaxis abstracta  
simplificado



WHILE LP VAR G\_T N RP LB ε PRINT VAR SEMIC VAR ASSIG VAR MINUS N SEMIC RB

while ( n > 0 ) { print n ; n = n - 1 ; }