

6.7 Consider the following grammar for simple Pascal-style declarations:

$\text{decl} \rightarrow \text{var-list} : \text{type}$   
 $\text{var-list} \rightarrow \text{var-list}, \text{id} \mid \text{id}$   
 $\text{type} \rightarrow \text{integer} \mid \text{real}$

Write an attribute grammar for the type of a variable.

grammar	semantic
$\text{decl} \rightarrow \text{var-list} : \text{type}$	$\text{var-list.dtype} = \text{type.dtype}$
$\text{var-list}_1 \rightarrow \text{var-list}_2, \text{id}$	$\text{id.dtype} = \text{var-list}_1.dtype$ $\text{var-list}_2.dtype = \text{var-list}_1.dtype$
$\text{var-list} \rightarrow \text{id}$	$\text{id.dtype} = \text{var-list.dtype}$
$\text{type} \rightarrow \text{integer}$	$\text{type.dtype} = \text{integer}$
$\text{type} \rightarrow \text{real}$	$\text{type.dtype} = \text{real}$

6.8 Consider the grammar of Exercise 6.7. Rewrite the grammar so that the type of a variable can be defined as a purely synthesized attribute, and give a new attribute grammar for the type that has this property.

$\text{decl} \rightarrow \text{id var-list}$   
 $\text{var-list} \rightarrow , \text{id var-list} \mid : \text{type}$   
 $\text{type} \rightarrow \text{integer} \mid \text{real}$

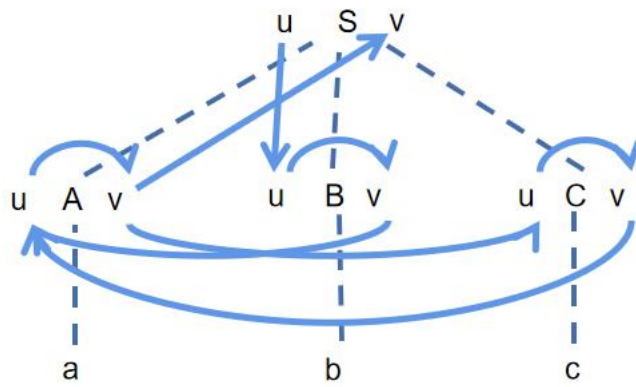
grammar	semantic
$\text{decl} \rightarrow \text{id var-list}$	$\text{id.dtype} = \text{var-list.dtype}$
$\text{var-list}_1 \rightarrow , \text{id var-list}_2$	$\text{id.dtype} = \text{var-list}_2.dtype$ $\text{var-list}_1.dtype = \text{var-list}_2.dtype$
$\text{var-list} \rightarrow : \text{type}$	$\text{var-list.dtype} = \text{type.dtype}$
$\text{type} \rightarrow \text{integer}$	$\text{type.dtype} = \text{integer}$
$\text{type} \rightarrow \text{real}$	$\text{type.dtype} = \text{real}$

6.13 Consider the following attribute grammar:

Grammar Rule	Semantic Rules
$S \rightarrow ABC$	$B.u = S.u$ $A.u = B.v + C.v$ $S.v = A.v$
$A \rightarrow a$	$A.v = 2 * A.u$
$B \rightarrow b$	$B.v = B.u$
$C \rightarrow c$	$C.v = 1$

- a. Draw the parse tree for the string **abc** (the only string in the language), and draw the dependency graph for the associated attributes. Describe a correct order for the evaluation of the attributes.

What value does  $S.v$  have after attribute evaluation, if  $S.u = 3$  before evaluation begins?



There is no meaningful value since the dependency is cyclic.