Binding Theory Quiz Answers

Jean Mark Gawron

Linguistics 522
San Diego State University
gawron@mail.sdsu.edu
http://www.rohan.sdsu.edu/~gawron

Antecedent

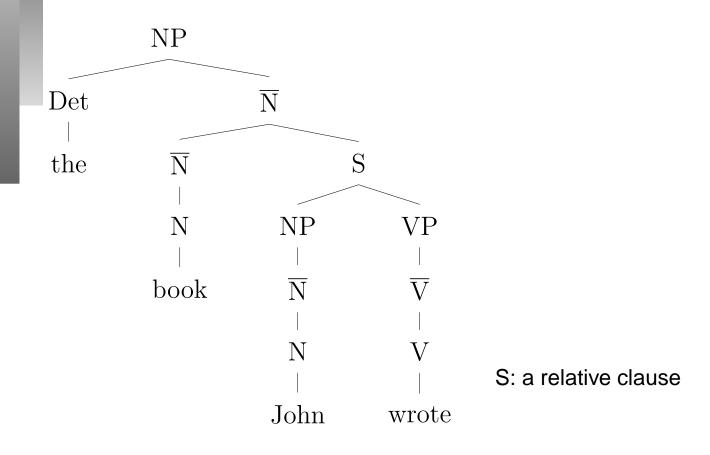
Assume the following definition of antecedent:

The **antecedent** of a pronoun or an anaphor is an R-expression coindexed with it.

For example, *John* is the antecedent of *he* and *himself* in:

- (i) John $_i$ thinks he $_i$ is a genius.
- (ii) John_i likes himself_i a lot.

Assume relative clauses are adjuncts of nouns

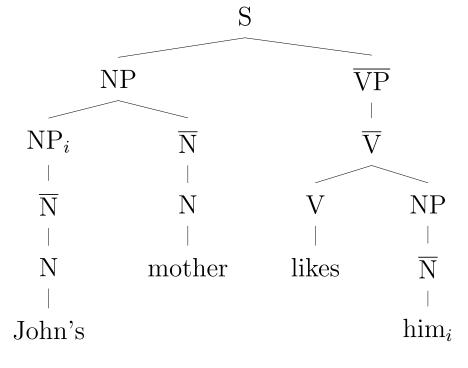


General directions

Answer true or false for each of the following. Grammaticality judgments for examples have been omitted because they are not relevant to answering any of the questions (and to make it clear that the answers do not DEPEND on the grammaticality judgments).

Pronoun antecedents

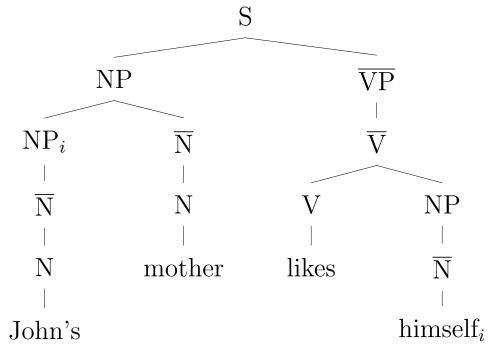
1. The antecedent of a pronoun must bind it. False



A non-binding antecedent. John's does not C-command him.

Anaphor antecedents

2. The antecedent of an anaphor must bind it. **True**



Impossibility of a non-binding antecedent

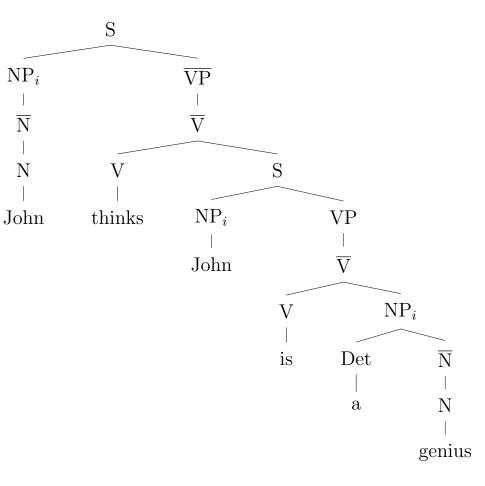
Principle C

The binding theory rules out the following sentence. True

N

 $John_i$ thinks $John_i$ is a genius.

The example in (3) is a Principle C violation. True John Ccommands and is coindexed with second John.



More True/False!

5. The binding theory rules out the following sentence **True**

He_i thinks John_i is a genius.

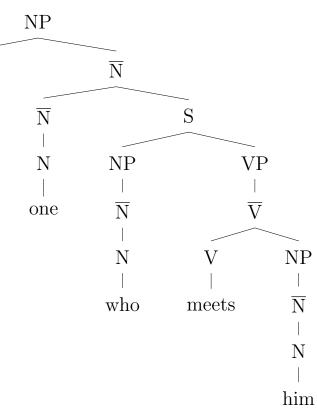
6. The example in (5) is a Principle C violation. True

Principle C Revisited

7. The binding theory rules out every the following sentence **False**. him cannot C-command anything outside its clause!

Det

Everyone who meets him_i admires $John_i$.



Binding theory decisions

8. Determine which of the following sentences the binding theory rules out and if a sentence is ruled out, state which principle(s) is/are involved.

(a)	Everyone who meets John $_i$ admires him $_i$.	In	
(b)	Every farmer who owns a donkey $_i$ beats it $_i$.	In	
(c)	Every farmer who owns it $_i$ beats a donkey $_i$.	In	
(d)	$Himself_i$ likes $John_i$.	Out	Prin C
			Prin A
(e)	$Himself_j$ likes $John_i$.	Out	Prin A
(f)	He_i likes $John_i$.	Out	Prin C

John_i likes her_j picture of him_i.

(g)

Prin B

Out