Ling 165B: Syntax II

Nico(letta) Loccioni

January 03, 2022

Constituency

Constituency and Constituency Tests I

One of the fundamental discoveries
about the syntax of natural languages is that languages are *chunky*: words are organized into chunks or blocks of units that 'rules' can manipulate as blocks.

→ Languages don't just consist of strings of words. They have syntactic structure.

→ Linguists use tree diagrams to represent the structure of languages.

Constituent: String (word or group of words) that speakers can manipulate as a single chunk (because they form a unit).

Constituency and Constituency Tests II

How do we identify the parts of sentences that form units (=the constituents)?

We use constituency tests!

Constituency Tests are diagnostics used to identify sentence structure.

→ To correctly use a constituency test, we must follow rules

→ First we perform a test, and then we check whether the result is:

(i) grammatical

(ii) consistent in meaning

→ If it is, than the string is a constituent.

→ Otherwise, we conclude nothing.

Replacement

How it works:
Given a well-formed string S that we are trying to analyze,
1. select a substring SUB
2. replace SUB in 'kind' by (what looks like) a monomorphic word (a word with no internal structure)
3. if the result R is well formed, we conclude that SUB is a constituent
4. If the result is ill formed, we conclude nothing

(1) S: The UPS courier$_{\text{SUB}}$ will deliver that big yellow package on Monday. → R: **He** will deliver that big yellow package on Monday.

S = The UPS courier will deliver that big yellow package on Monday
SUB = The UPS courier
R = He will deliver that big yellow package on Monday.

Logic of the test:

- The smallest possible constituent consists of a single word
- Any string of words that can be replaced by a single word while maintaining the same meaning is a constituent

Remember:

- Original string and replacement should be minimal pairs
- S and R have to be very close synonym.

What can we infer from a successful substitution?

- The initial phrase is a constituent
- The initial phrase and its substitution share a distributional property (i.e. they belong to the same category).
- In the tree structure, SUB is a subtree of S.

⇒ **Replacement by 'one"**

Example

(2) The UPS courier will deliver that big yellow $\underline{\text{package}}_{\text{SUB}}$ on Monday → The UPS courier will deliver that big yellow **one** on Monday. ✓

This shows that 'package' forms a constituent in (2).

(3) That big $\underline{\text{yellow}}_{\text{SUB}}$ package → *That big **one** package

(4) That big yellow package $\underline{\text{arrived}}_{\text{SUB}}$ → *That big yellow package **one**

What kinds of constituents can 'one' replace?

It can replace nominal phrases, NPs.

From the test in (2), we can infer that $[_{\text{NP}}$ package]

(5) The UPS courier will deliver that big $\underline{\text{yellow package}}_{\text{SUB}}$
→ The UPS courier will deliver that big **one** ✓

From the test in (5), we can infer $[_{\text{NP}}$ yellow package]

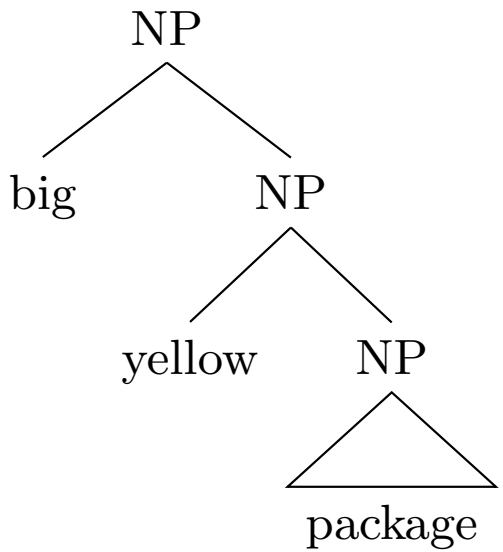
(6) The UPS courier will deliver that <u>big yellow package</u>$_{\text{SUB}}$
→ The UPS courier will deliver that **one** ✓

From the test in (6), we can infer that [$_{\text{NP}}$ big yellow package]

We can therefore infer the following structure:

[$_{\text{NP}}$ big [$_{\text{NP}}$ yellow [$_{\text{NP}}$ package]]]

which can be represented as follows:



⇒ Replacement by a pronoun

Examples

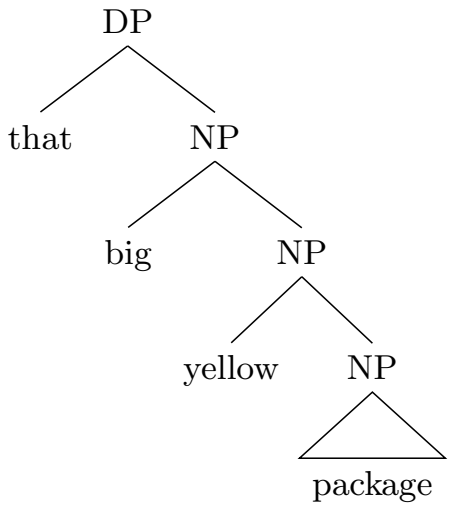
(7) Bill$_{\text{SUB}}$ loves kayaking. → **He** loves kayaking.

(8) Jim and Susan$_{\text{SUB}}$ left. → **They** left.

(9) The UPS courier will deliver that big yellow package$_{\text{SUB}}$ on Monday
→ The UPS courier will deliver **it** on Monday ✓

(10) a. That big yellow package$_{\text{SUB}}$ arrived → *That **it** arrived
b. That big yellow package$_{\text{SUB}}$ arrived → **It** arrived ✓

What kinds of constituents can a pronoun replace?



It replaces DPs. It can't replace bare nominal phrases, NPs.

⇒ **Replacement by 'there/then'**

Example

(11) I will put the book <u>on the table</u>$_{\text{SUB}}$. → I will put the book **there**.

(12) The screening will be <u>at the movie theater</u>$_{\text{SUB}}$. → The screening will be **there**.

(13) The UPS courier will deliver that big yellow package <u>on Monday</u> → The UPS courier will deliver that big yellow package **then**.

What kinds of constituents can *there/then* replace?

It can replace PPs.

⇒ **Replacement by 'do so'**

Example

(14) The UPS courier will deliver that big yellow package on Monday → The UPS courier will do so on Monday.

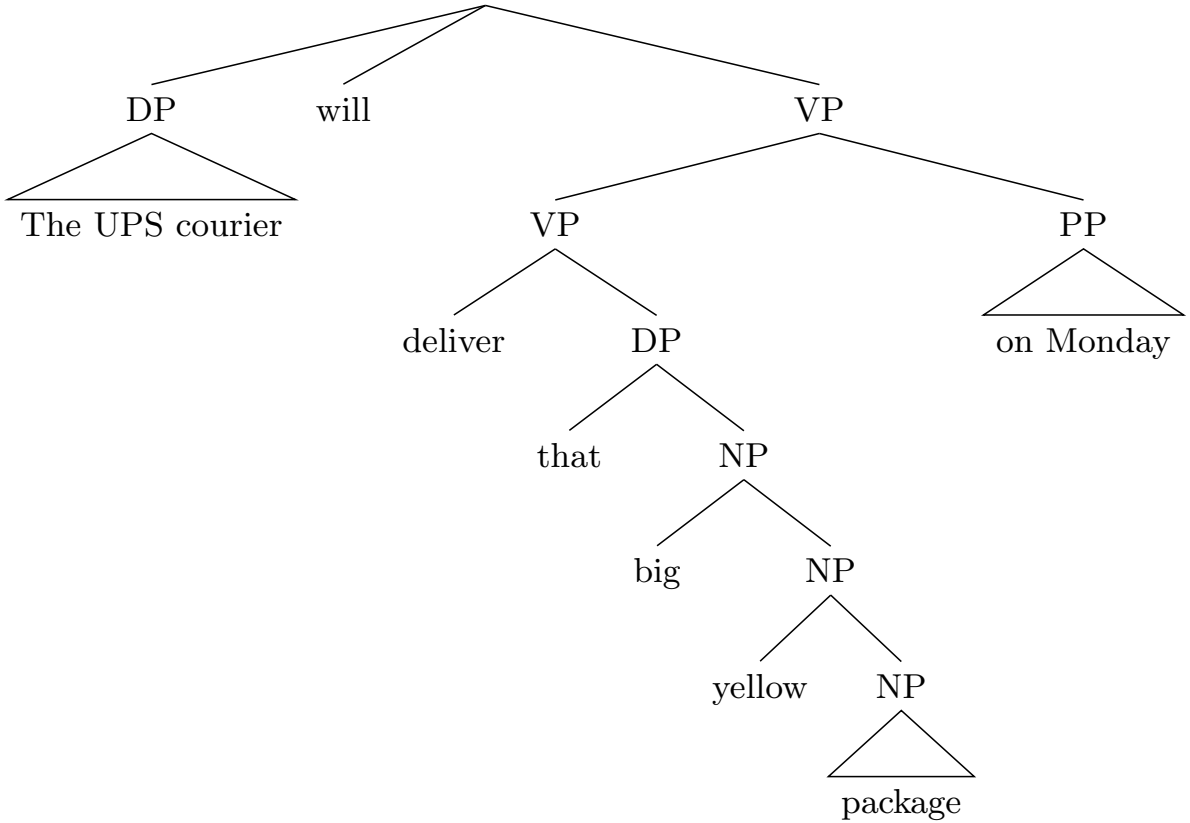
(15) The UPS courier will deliver that big yellow package on Monday → The UPS courier will do so.

(16) John usually walks to school$_{\text{SUB}}$ → John usually **does so**

What kinds of constituents can 'do so' replace?

It can replace VPs.

The constituency tests we ran gave us evidence for the following tree structure for *The UPS courier will deliver that big yellow package on Monday*:

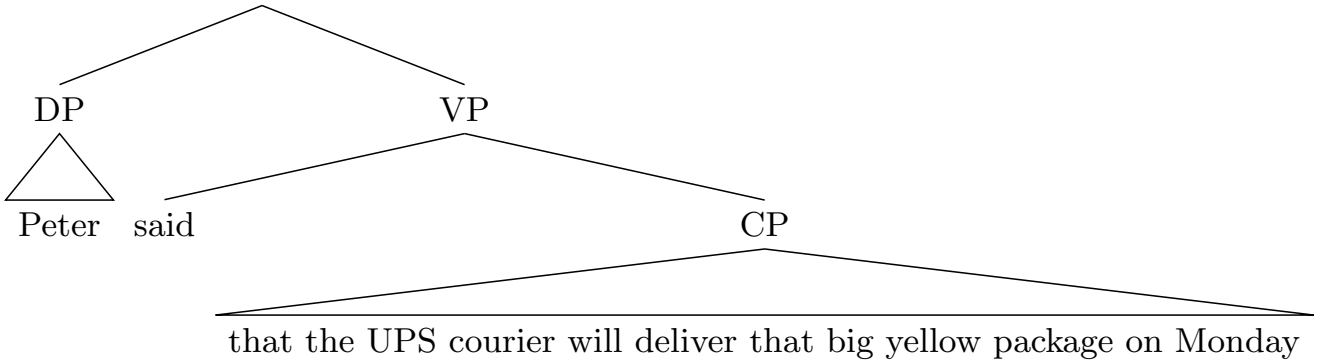


⇒ **Replacement by 'so'**

You can embed the sentence above:

(17) Peter said
that the UPS courier will deliver that big yellow package on Monday→ Peter said **so**

We will label this type of constituent CP.



Summary: Replacement tests

What to remember:

• When carefully used replacement can tell us about the constituency of sentences
• A successful replacement shows that the replaced string is a syntactic unit, i.e. constituent
• In terms of trees:

- A constituent is a node in the tree
- Constituency tests reveal the major shape of the tree

Coordination

Coordination Test:
Take two acceptable sentence of the form

[A B D] and [A C D]

where A, B, C and D represent (possibly null) substrings.
If the string [A B *and* C D] is acceptable with the same meaning
as [A B D *and* A C D], this is evidence that B and C are both constituents, and constituents of the same type.

Example

(18) a. John will send [$_B$ a card] to his mother.
b. John will send [$_C$ a present] to his mother.

(19) John will send [$_B$ a card] to his mother *and* John will send [$_C$ a present] to his mother.

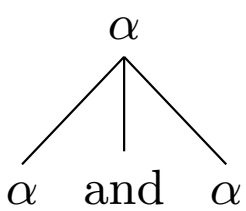
(20) John will send [$_B$ a card] and [$_C$ a card] to his mother

We conclude that *a card* and *a present* are constituents of the same kind.

The coordination test assumes that only constituents can be coordinated and, following the coordination rule only identical constituents can be coordinated!

Coordination rule:

(21) $\underline{\alpha \rightarrow \alpha \textbf{ and } \alpha}$

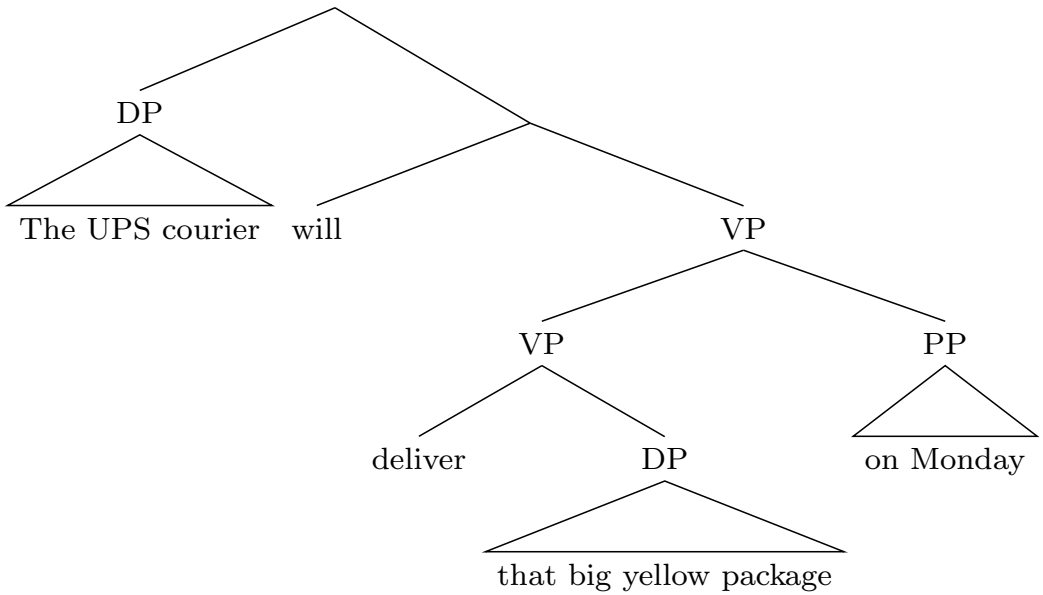


[α: variable over NP, N, A, V, VP, Det, PP, TP...]

With this test, we can show that the auxiliary (or the modal) form a constituent with the VP to the exclusion of the subject:

(22) The UPS courier [will deliver that big yellow package on Monday] and [should do it by 8pm].

Let's update our tree!



→ Topicalization (or fronting)

Topicalization allows one to move a DP, PP or VP constituent to the front of a sentence. The moved constituent is called the (contrastive) topic.

Thus, topicalization can only affect constituents (but not all of them!).

(23) John gave tuna to his cat.
   a. Tuna, John gave < *tuna* > to his cat.
   b. His cat, John gave tuna to <*his cat* >
   c. To his cat, John gave tuna < *to his cat* >.
   d. Give tuna to his cat, John did < *give tuna to his cat* >

Consider those failed attempts at topicalization. What can we conclude from them?

(24) John gave tuna to the cat of his neighbor.
   a.*His neighbor, John gave tuna to the cat of <*his neighbor* >.

→ Clefting

Clefting is another construction where a constituent gets shuffled around. In this case, the affected constituent is called the focus.

Schematically, a cleft looks like this:
A B C ⇒ It's B that A C.

(25) a. It's tuna that John gave <*tuna*> to his cat.
b. It's to his cat that John gave tuna <*to his cat*>.
c. It's his cat that John gave tuna to < *his cat*>.
d. It's John that <*John*> gave tuna to his cat.

If the result is acceptable, we have evidence that the string of words in focus form a constituent.

While VPs can be topicalized, they cannot be clefted.

(26)*It's give tuna to his cat that John did <*give tuna to his cat*>.

→ Pseudoclefts

Pseudoclefting is very similar to clefting in what it does to the information structure of the sentence.

A B C ⇒ What A C is B.

(27) What John gave <*tuna*> to his cat is tuna.

Pseudoclefts using a wh- word other than 'what' are judged marginal by most speakers. Because of this, 'John' is a difficult target.

(28) ?Who <*John*> gave tuna to his cat is John.

In most cases, PPs simply cannot be pseudoclefted.

In addition, something we probably want to call a VP pseudocleft is possible. (A B C ⇒ What A do C is B.)

(29) What John did <*give tuna to his cat*> is give tuna to his cat.

VP-ellipsis (VPE)

In many languages including English, a non-finite verb phrase can be left out if its antecedent is contextually provided:

(30) Daniel eats meat. He shouldn't.

Also common in tag questions:

(31) Daniel eats meat, doesn't he?

This phenomenon can be use as a test: in order for a string to be the target of ellipsis, it has to form a constituent.

(32) a. Chris will write a new poem
b. Chris will.

For speakers of a language other than English, do you have VP ellipsis in your language? I don't in Italian.

(33) *Adriana sta leggendo un libro e Piero sta anche.
Adriana is reading a book and Piero is too.

NP-ellispis (NPE)

It is also possible to omit the NP is English? Let's try to figure out in which contexts a NP can be omitted in English.

(34) a. When Peter brings his cute dog, John brings his, too.
b. Because you brought two donuts, I brought two.
c. Some school kids like syntax, and some school kids don't

Practice Draw a tree representation for the following sentence. For each constituent you draw, provide one experiment (constituency test) justifying it. Pair up the constituency test result with the node in the tree that the test give evidence for.

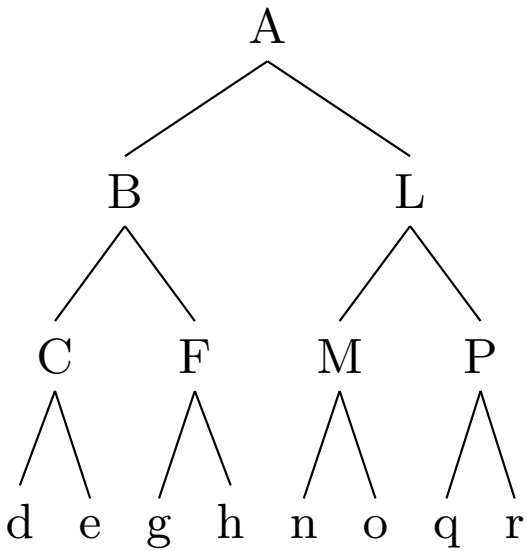
(35) That kind French guy will quickly bake a cake.

Structural relations

Branch
A line connecting two parts of a tree.

Node
Each point that is labeled with a word or a category is called a *node*. It is the end of a branch.

Label
The name given to a node.



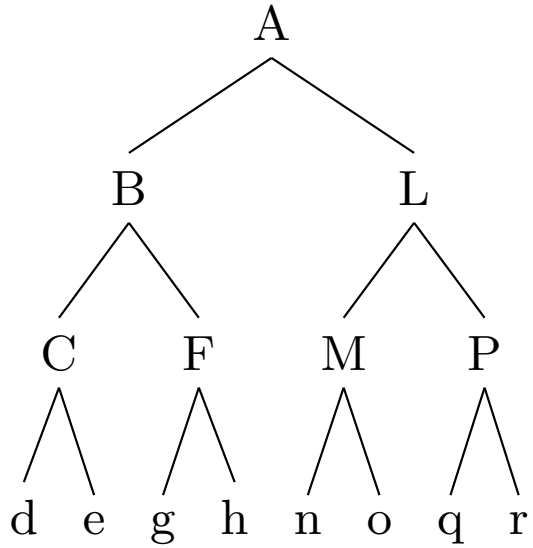
Dominance
A node α dominates a node β iff there exists a chain of two or more nodes $\alpha, \gamma_i, ..., \gamma_j, \beta$ such that each node is the mother of the next one. **(Roughly, α is an ancestor of β.)**

Root
The node that dominates all other nodes in a tree, and is itself dominated by none, is called the root *node*.

Leaf (or terminal node)
The nodes along the bottom of the tree are called *leaves*. They dominate nothing.

Non-terminal node
A node that dominates something. (A node that is a mother).

Immediate dominance
A node α immediately dominates a node β iff (i) α dominates β, and (ii) there is no node $\gamma \neq \alpha$ such that γ dominates β. **(In other words, α is β's mother.)**



Mother
A is the mother of B if A immediately dominates B.

Daughter
B is the daughter of A if B is immediately dominated by A.

Sisters
Two nodes that share the same mother.

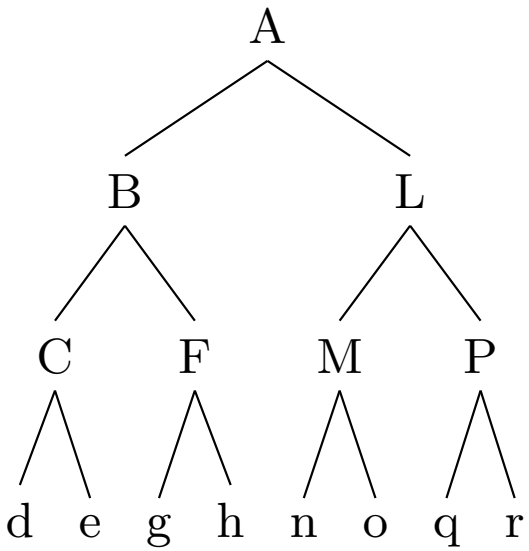
Sister Precedence
Node A sister-precedes node B if and only if both are immediately dominated by the same node, and A appears to the left of B.

Precedence

Node A precedes node B if and only if

(i) neither A dominates B nor B dominates A and

(ii) A (or some node dominating A) sister precedes B (or some node dominating B)



C(onstituent)-Command:

Node A c-commands node B if

(i) every node dominating A also dominates B;

(ii) and A does not itself dominate B

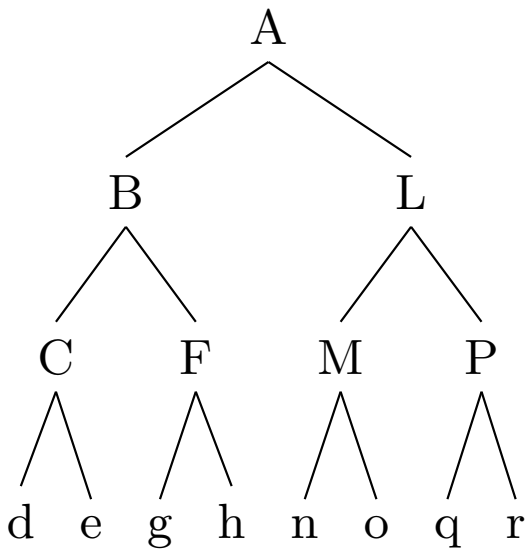
Asymmetric C-Command:

A asymmetrically c-commands B if A c-commands B but B does NOT c-command A.

Exhaustive dominance

A node α exhaustively dominates a set of terminal nodes N iff (i) α dominates every node in N (so that there is no member of the set N that is not dominated by α), and (ii) α does not dominate any terminal node not in N.

- Exhaustive dominance is useful to talk about the leaves of a tree (which are the lexical items).
- Leaves form a *constituent* precisely when there is a node in the tree to exhaustively dominate those leaves.

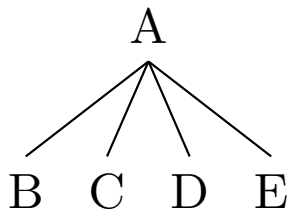


NOTA BENE: immediate domination is not a necessary condition for exhaustive dominance.

In tree (36), A exhaustively dominates B, C, D and E because:

- B, C, D and E are terminal nodes
- A dominates all of them
- there is no other terminal node dominated by A (i.e B, C, D and E are the only terminal nodes dominated by A.

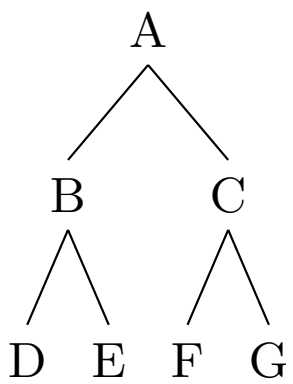
(36)



In structure (37), A exhaustively dominates D, E, F, G because:

- D, E, F and G are terminal nodes
- A dominates all of them (it does not matter that it does not immediately dominate them)
- there is no other terminal node dominated by A (i.e D, E, F and G are the only terminal nodes dominated by A.

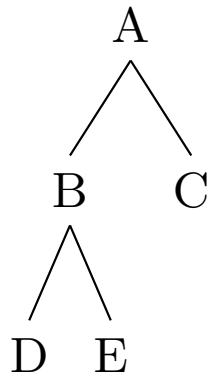
(37)



In structure (38), A exhaustively dominates D, E and C but it does not exhaustively dominate D and E, because:

- it is true that D and E are terminal nodes
- it is also true that A dominates all of them (it does not matter that it does not immediately dominate them)
- but there is another terminal node (namely C) which is dominated by A.

(38)



Practice Draw a tree with the following properties:

- non-binary branching nodes are forbidden;
- A is a root node;
- f, g, h, i, l and m and g are all terminal rodes;
- A, B, C, D and E are not-terminal nodes;
- B sister-precedes E;
- D asymmetrically c-commands B;
- D sister-precedes C;
- A does not immediately dominate B;
- D exhaustively dominates { m, l }
- m precedes f;
- i precedes g;
- l sister-precedes m;
- g is not i's sister;
- E dominates h.
- i does not sister precede f
- g precedes h

Structural ambiguities

A sentence can have more than one meaning. It's called 'ambiguity'

It comes in two flavors:

(i) **Lexical ambiguity** uses words with more than one meaning.

(39) Peter went to the bank.
   a. ≡ Peter went to the financial institution.
   b. ≡ Peter went to the side of the river.

There are two lexical entries for 'bank': $bank_1$ and $bank_2$.

(ii) **Structural ambiguity** ambiguous due to the structure or constituency of the sentence.

The relationship between structure and meaning is systematic!

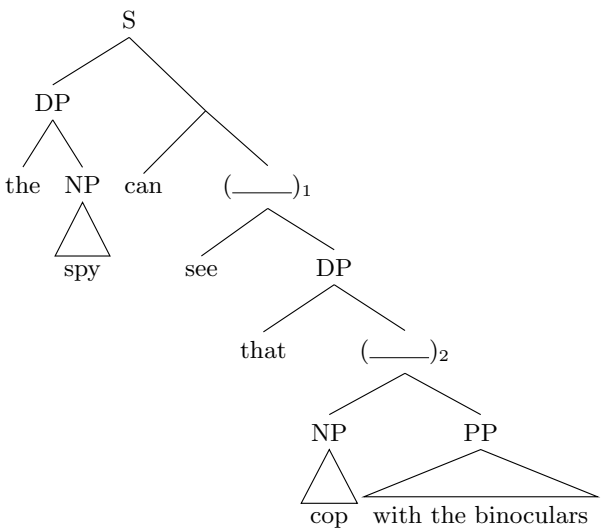
Example

(40) The spy can see that cop with the binoculars.

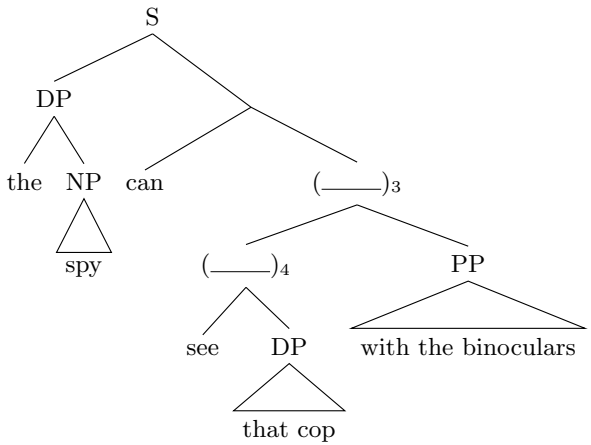
a. ≡ _____

b. ≡ _____

Tree 1



Tree 2



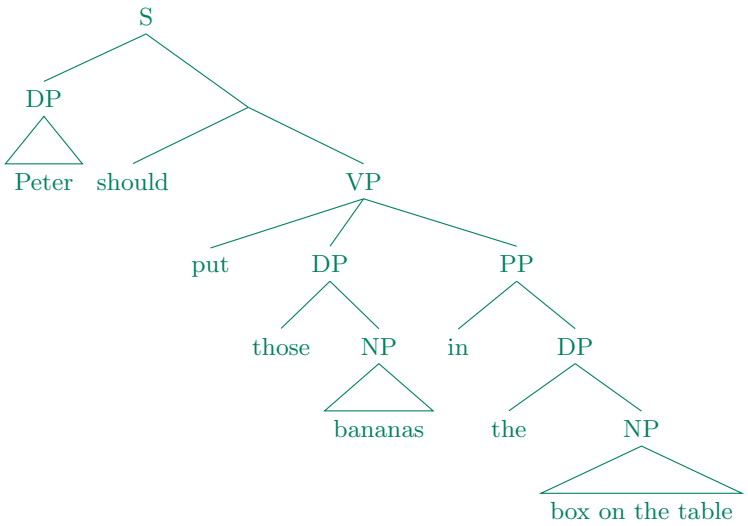
NB: Each tree has an unambiguous interpretation!

- Pair each tree up with the meaning of each sentence;
- Label the nodes 1,2,3 and 4;
- Explain how at least one of the tests supports the structure in the first tree but not the second and how at least one test supports the structure in the second tree but not in the first.

For you to practice!

Consider the following ambiguous sentence:

(41) Peter should put those bananas in the box on the table.



- Draw the other tree structure.
- Pair each tree up with the meaning of each sentence;
- Explain how at least one of the tests supports the structure of one tree but not the other.

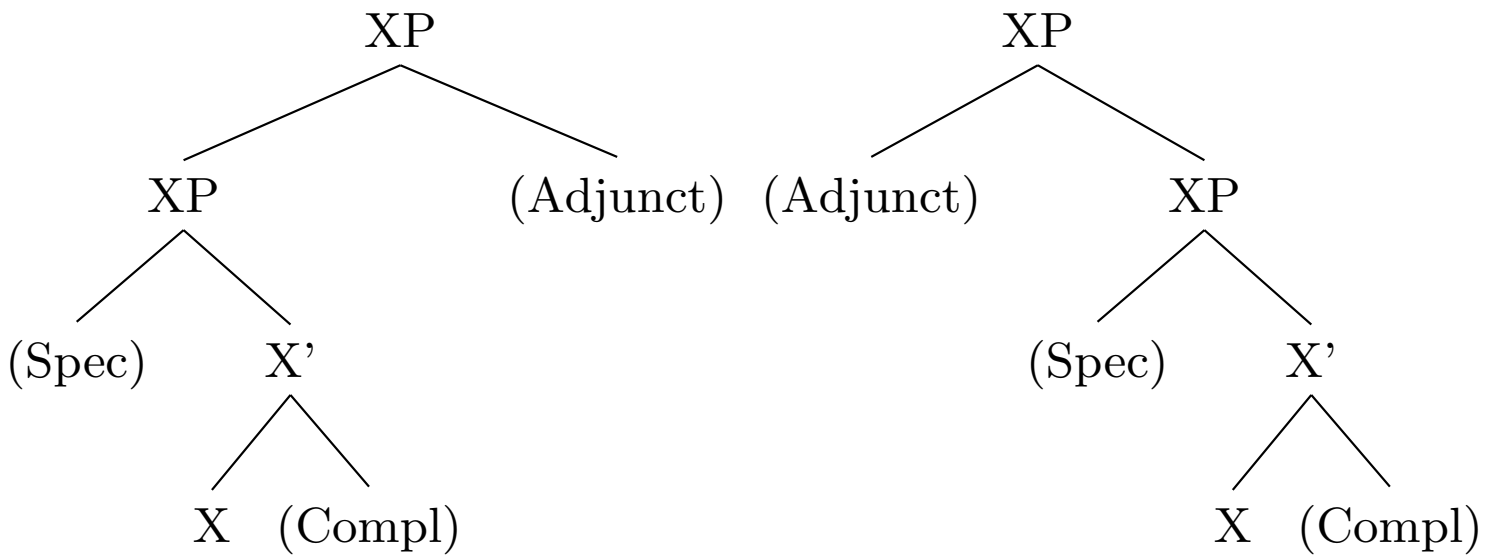
Coming up: X-bar theory

- We know that sentences exist! (So far, we have been using the label 'S' for sentences)
- Based on constituency tests, we have been able to identify:
  - Verb Phrases (VPs);
    do so replacement test; topicalization; pseudo-clefting; coordination; VPE, ...
  - Determiner Phrases (DPs);
    replacement using a pronoun; topicalization; clefting; pseudo-clefting; coordination ...
  - Noun Phrases (NPs);
    substitution by 'one(s)'; coordination; ...
  - Prepositional Phrases (PPs);
    replacement test (there, then...); topicalization; clefting; coordination: NPE...

  We also came across other parts of speech (adjectives, adverbs), modal verbs (should, will, must, can...)

Behind the range of diverse constructions that English and other languages allow, we find surprising uniformity and regularity.

- Next time we'll introduce the idea that the rule system that undelies our phrases is very simple.
- Every phrase looks the same!
- For English, syntax will be the iteration of:



This is called **X-bar Theory**.