LING/C SC 581:
Advanced Computational Linguistics

Lecture 25

Last Time

Language $\{a^n b^n c^n \mid n>0\}$ is not context-free

- Three implementations using Prolog grammar rules:
  1. CFG (context-free grammar) + extra arguments for grammatical constraints
  2. CFG + counting, cf. Perl
  3. CSG (context-sensitive grammar) rules

Next week

- Two special lectures to replace lectures 26 and 27, co-located with Simin's class
  - LING 506: *Major Works in Syntactic Theory*
- Location and Time:
  - Soc Sci 307, 12:30-1:45pm 19th and 21st
- Background reading:
  - section 5 of this paper
  - *Simple Models*: *Computational and Linguistic Perspectives*, Fong, S. (2022).
  - available for download on course website

Today's Topic

- Prolog grammar writing for natural languages
- Homework 11
- Some *live* programming to get you started …

Homework 11

Has two parts.

- Question 1: due next Monday midnight.
- Question 2: due the following week (Monday after that).

Homework 11: Question 1

- Write a Prolog phrase structure grammar for sentences:
  - John won the race.
  - Who won the race.
  - What did John win?
  - John won what? (ok as an echo-question)
- **Notes**:
  - You can ignore/discount the punctuation.
  - You should use the copy theory of movement: WHNP … WHNP
    - not Trace theory: WHNP … *trace*
  - You do not need ROOT (*see next slide*).

Homework 11: Question 1

- Recall the Standalone Stanford Parser from lecture 15.

```
(ROOT
  (S
    (NP (NNP John))
    (VP (VBD won)
      (NP (DT the) (NN race)))
    (. .)))
```

```
(ROOT
  (SBARQ
    (WHNP (WP Who))
    (SQ
      (VP (VBD won)
        (NP (DT the) (NN race))))
    (. .)))
```

```
(ROOT
  (SBARQ
    (WHNP (WP What))
    (SQ (VBD did)
      (NP (NNP John))
      (VP (VB win)))
    (. ?)))
```

```
(ROOT
  (S
    (NP (NNP John))
    (VP (VBD won)
      (NP (WP what)))
    (. ?)))
```

Homework 11: Question 1

- Equivalent Prolog parses (with copy theory).

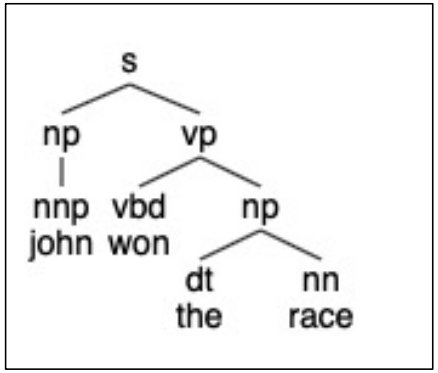
```
s(np(nnp(john)), vp(vbd(won), np(dt(the),
nn(race))))
  (S
    (NP (NNP John))
    (VP (VBD won)
      (NP (DT the) (NN race))))
```

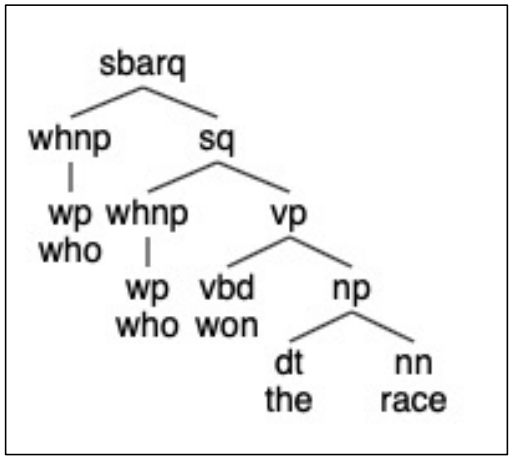
```
sbarq(whnp(wp(who)), sq(whnp(wp(who)),
vp(vbd(won), np(dt(the), nn(race)))))
  (SBARQ
    (WHNP (WP Who))
    (SQ
      (VP (VBD won)
        (NP (DT the) (NN race)))))
```

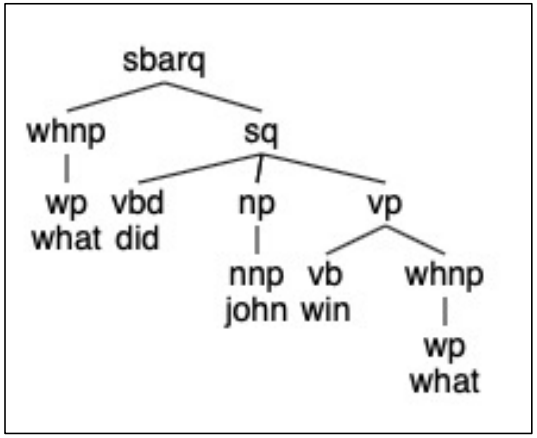
```
sbarq(whnp(wp(what)), sq(vbd(did),
np(nnp(john)), vp(vb(win), whnp(wp(what)))))
  (SBARQ
    (WHNP (WP What))
    (SQ (VBD did)
      (NP (NNP John))
      (VP (VB win))))
```

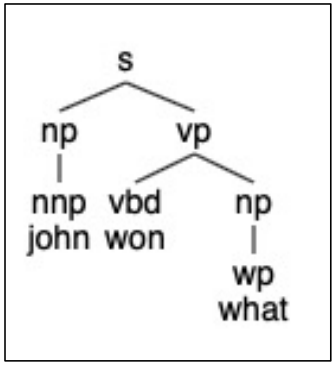
```
s(np(nnp(john)), vp(vbd(won), np(wp(what))))
  (S
    (NP (NNP John))
    (VP (VBD won)
      (NP (WP what))))
```

Homework 11: Question 1









Homework 11: Question 1

- Example:
  - `parse(Parse,[john, won, the, race], []).`
- Notes:

  Capitalization:
  - `John` would be a Prolog variable (begins with a capital letter)
  - `'John'` or `john` would be a Prolog atom

  Start symbol:
  - `parse(Tree) --> s(Tree).`
  - `parse(Tree) --> sbarq(Tree).`

  Copy theory:
  - use an extra argument (with a variable) to pass the copied phrase around the grammar rules
  - for a missing/unspelled subject/object, use a rule with an empty RHS
  - e.g. `nonterminal --> [].`

Homework 11: Question 2

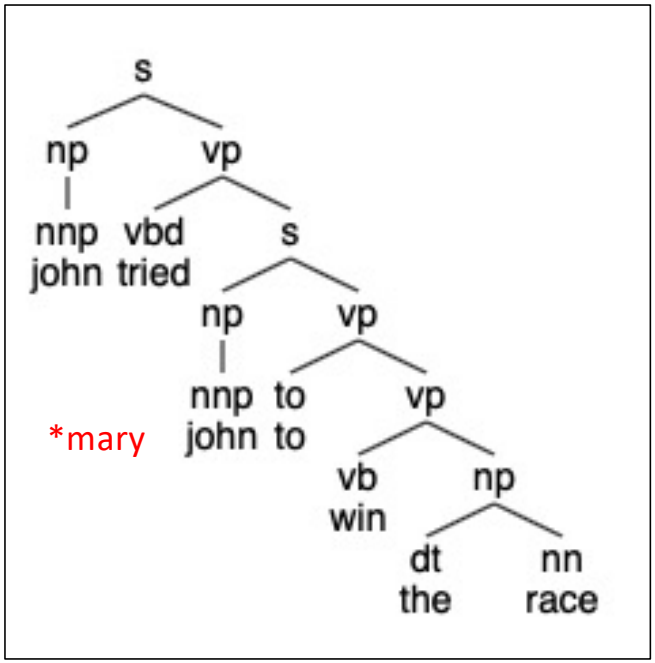
Extend your Prolog phrase structure grammar to handle sentences:

- John tried to win the race. (*subject control*)
- *John tried Mary to win the race.
- Who tried to win the race?
- What did John try to win?
- *Who John tried to win the race?
- *Who did John try to win the race?

- **Note**:
  - try to handle as many examples (from above) as you can
  - try to keep your rules as "elegant"/"simple" as you can
  - for subject control, use John … John

Homework 11: Question 2

```
(ROOT
  (S
    (NP (NNP John))
    (VP (VBD tried)
      (S
        (VP (TO to)
          (VP (VB win)
            (NP (DT the) (NN race))))))
    (. .)))
```




```
s(np(nnp(john)), vp(vbd(tried),s(np(nnp(john)),vp(to(to),vp(vb(win),np(dt(the),nn(race)))))))
```