1. Project questions?
2. Regular vs. context-free languages
3. Grammar ambiguity
4. (fun) Grammars for NLP
Regular or Context-Free? #1

- Reminder: $0^n$ means $n$ zeros
- $L = \{0^n1^n \mid n \geq 0\}$

$$E \rightarrow 0 \ E \ 1$$

$$E \rightarrow \varepsilon$$
Regular or Context-Free? #2

- $L = \{0^n1^m \mid n \geq 0, m > n\}$

$E \rightarrow 0\ E_1\ 1$
$E \rightarrow \ E_1$
$E_1 \rightarrow \ E'\ 1$
$E_1 \rightarrow \ E_1\ 1$
$E' \rightarrow \ E$
$E' \rightarrow \ \varepsilon$
Regular or Context-Free? #3

• $\#<x>$ counts occurrences of substring $x$
• $L = \{ w \text{ s.t. } |\#00(w) - \#11(w)| \leq 3 \}$
Regular or Context-Free? #3

• \(#<x>\) counts occurrences of substring \(x\)

• \(L = \{w \text{ s.t. } |\#00(w) - \#11(w)| \leq 3\}\)
Regular or Context-Free? #5

• \(\#<x>\) counts occurrences of substring \(x\)
• \(L = \{ w \mid \#0(w) == \#1(w) \}\)

\[
E \rightarrow \varepsilon \mid 0 \ E_1 \mid 1 \ E_0 \\
E_1 \rightarrow 1 \ E \mid E \ 1 \\
E_0 \rightarrow 0 \ E \mid E \ 0
\]
Regular or Context-Free? #6

• Notation: \( \#<x> \) counts occurrences of substring \( x \)

• \( L = \{w \mid \#01(w) = \#10(w)\} \)

\[(0+(1+0+)^*|1+(0+1+)^*)^?\]
Some Things That Aren’t Regular

• Counting to infinity
• Duplication of arbitrary-length strings
Ambiguous Grammars #1

- Example from class: calculator (eg 5+6*7)
  - Solve by changing the grammar

\[
\begin{align*}
expr &::= expr + expr \mid expr - expr \\
    &\quad \mid expr \times expr \mid expr / expr \mid int \\
int &::= 0 \mid 1 \mid 2 \mid 3 \mid 4 \mid 5 \mid 6 \mid 7 \mid 8 \mid 9
\end{align*}
\]

\[
\begin{align*}
expr &::= expr + term \mid expr - term \mid term \\
term &::= term \times factor \mid term / factor \mid factor \\
factor &::= int \mid ( expr ) \\
int &::= 0 \mid 1 \mid 2 \mid 3 \mid 4 \mid 5 \mid 6 \mid 7
\end{align*}
\]
Ambiguous Grammars #3

• >> in C++

```cpp
list<list<int>> l;  // generics
int i = 318491 >> 5;  // right shift
```

• Grammar for this?
Ambiguous Grammars #3

• >> in C++

    list<list<int>> l;                           generics

    int i = 318491 >> 5;                         right shift

    template <const int N> class factorialT {...}
    list<factorialT<5>> l;                    compile-time computation

    template <const int N> class factorialT {...}
    list<factorialT<318491 >> 5>> l;         ???

• How to avoid an ambiguous grammar?
• Do we want a nonambiguous grammar?
Natural Languages are Ambiguous

- First try at a cfg for English:

  S → NP VP
  NP → DET NP | NN | NP PP
  VP → V NP | VP PP
  PP → P NP
  N → boy | girl | house | UFO | radar | telescope | alien | government conspiracy
  V → sees | meets | leaves | knows | greets
  DET → a | an | the
  P → to | for | with | in
Natural Languages are Ambiguous

• First try at a cfg for English:

\[
\begin{align*}
S & \rightarrow NP \ VP \\
NP & \rightarrow DET \ NP \mid NN \mid NP \ PP \\
VP & \rightarrow V \ NP \mid VP \ PP \\
PP & \rightarrow P \ NP \\
N & \rightarrow \text{boy} \mid \text{girl} \mid \text{house} \mid \text{UFO} \mid \text{radar} \mid \text{telescope} \mid \text{alien} \mid \text{government conspiracy} \\
V & \rightarrow \text{sees} \mid \text{meets} \mid \text{leaves} \mid \text{knows} \mid \text{greets} \\
DET & \rightarrow \text{a} \mid \text{an} \mid \text{the} \\
P & \rightarrow \text{to} \mid \text{for} \mid \text{with} \mid \text{in}
\end{align*}
\]

• How to parse?
  – the girl sees the boy in the telescope
Natural Languages are Not Context-Free

• Long-distance dependences
  – Who does the roll call?
  – For whom does the roll call?

• Context-sensitive verbs
  – I’d do what he would
  – I’d do what he would do
  – I’d run faster than he would
  – * I’d run faster than he would do