BILL AND NATHAN START RECORDING

Context Sensitive Languages

Why Are Context Sensitive Languages Important

Why Are Context Sensitive Languages Important

I am supposed to say
Context Sensitive Languages **are** *important in Linguistics.*

Why Are Context Sensitive Languages Important

I am supposed to say
Context Sensitive Languages **are** *important in Linguistics.*

It is more accurate to say

Why Are Context Sensitive Languages Important

I am supposed to say
Context Sensitive Languages **are** *important in Linguistics.*

It is more accurate to say

Context Sensitive Languages **were** *important in Linguistics.*

Why Are Context Sensitive Languages Important

I am supposed to say
Context Sensitive Languages **are** *important in Linguistics.*

It is more accurate to say

Context Sensitive Languages **were** *important in Linguistics.*

Our interest in CFL's is:

Why Are Context Sensitive Languages Important

I am supposed to say
Context Sensitive Languages **are** *important in Linguistics.*

It is more accurate to say

Context Sensitive Languages **were** *important in Linguistics.*

Our interest in CFL's is:
1) Languages that require a LARGE CFG but a SMALL CSL

Why Are Context Sensitive Languages Important

I am supposed to say
Context Sensitive Languages **are** *important in Linguistics.*

It is more accurate to say

Context Sensitive Languages **were** *important in Linguistics.*

Our interest in CFL's is:
1) Languages that require a LARGE CFG but a SMALL CSL
(Research Project—no results yet.)

Why Are Context Sensitive Languages Important

I am supposed to say
Context Sensitive Languages **are** *important in Linguistics.*

It is more accurate to say

Context Sensitive Languages **were** *important in Linguistics.*

Our interest in CFL's is:
1) Languages that require a LARGE CFG but a SMALL CSL
(Research Project—no results yet.)
2) Closure properties of CSLs.

Why Are Context Sensitive Languages Important

I am supposed to say
Context Sensitive Languages **are** *important in Linguistics.*

It is more accurate to say

Context Sensitive Languages **were** *important in Linguistics.*

Our interest in CFL's is:
1) Languages that require a LARGE CFG but a SMALL CSL
(Research Project—no results yet.)
2) Closure properties of CSLs.
3) How does CSL compare to P and other classes?

Why Are Context Sensitive Languages Important

I am supposed to say
Context Sensitive Languages **are** *important in Linguistics.*

It is more accurate to say

Context Sensitive Languages **were** *important in Linguistics.*

Our interest in CFL's is:
1) Languages that require a LARGE CFG but a SMALL CSL (Research Project—no results yet.)
2) Closure properties of CSLs.
3) How does CSL compare to P and other classes? (Spoiler Alert: We don't know!)

Why Are Context Sensitive Languages Important

I am supposed to say
Context Sensitive Languages **are** *important in Linguistics.*

It is more accurate to say

Context Sensitive Languages **were** *important in Linguistics.*

Our interest in CFL's is:
1) Languages that require a LARGE CFG but a SMALL CSL (Research Project—no results yet.)
2) Closure properties of CSLs.
3) How does CSL compare to P and other classes? (Spoiler Alert: We don't know!)
4) Which languages are **not** context sensitive?

Why Are Context Sensitive Languages Important

I am supposed to say
Context Sensitive Languages **are** *important in Linguistics.*

It is more accurate to say

Context Sensitive Languages **were** *important in Linguistics.*

Our interest in CFL's is:
1) Languages that require a LARGE CFG but a SMALL CSL (Research Project—no results yet.)
2) Closure properties of CSLs.
3) How does CSL compare to P and other classes? (Spoiler Alert: We don't know!)
4) Which languages are **not** context sensitive? (Spoiler Alert: very few natural languages that are not CSL are known.)

Why Are Context Sensitive Languages Important

I am supposed to say
Context Sensitive Languages **are** *important in Linguistics.*

It is more accurate to say

Context Sensitive Languages **were** *important in Linguistics.*

Our interest in CFL's is:
1) Languages that require a LARGE CFG but a SMALL CSL
(Research Project—no results yet.)
2) Closure properties of CSLs.
3) How does CSL compare to P and other classes? (Spoiler Alert: We don't know!)
4) Which languages are **not** context sensitive? (Spoiler Alert: very few natural languages that are not CSL are known.)
5) Languages that are CSL but not CFL.

Historical Note on Linguistics

One of the motivations for CFL's and CSL's is an attempt to model human language.
This was a success and a success.

Historical Note on Linguistics

One of the motivations for CFL's and CSL's is an attempt to model human language.
This was a success and a success.

1. While human language is far more complicated than CFL or CSL; the Mathematical tools these grammars supply were a helpful starting point.

Historical Note on Linguistics

One of the motivations for CFL's and CSL's is an attempt to model human language.
This was a success and a success.

1. While human language is far more complicated than CFL or CSL; the Mathematical tools these grammars supply were a helpful starting point.
2. Computer languages are far easier to understand since we make them ourselves; hence, CFLs and (to a lesser extent) CSL's were useful within Computer Science.

Examples of Context Sensitive Grammars

$S \to ABCS \mid e$
$AB \to BA$ (Note- We allow two nonterminals on the LHS.)
$AC \to CA$
$BC \to CB$
$BA \to AB$
$CA \to AC$
$CB \to BC$
$A \to a$
$B \to b$
$C \to c$

Examples of Context Sensitive Grammars

$S \to ABCS \mid e$

$AB \to BA$ (Note- We allow two nonterminals on the LHS.)

$AC \to CA$

$BC \to CB$

$BA \to AB$

$CA \to AC$

$CB \to BC$

$A \to a$

$B \to b$

$C \to c$

1) What lang does this generate?

Examples of Context Sensitive Grammars

$S \to ABCS \mid e$
$AB \to BA$ (Note- We allow two nonterminals on the LHS.)
$AC \to CA$
$BC \to CB$
$BA \to AB$
$CA \to AC$
$CB \to BC$
$A \to a$
$B \to b$
$C \to c$
1) What lang does this generate?
$\{w : \#_a(w) = \#_b(w) = \#_c(w)\}$ which is NOT a CFL.

Examples of Context Sensitive Grammars

$S \to ABCS \mid e$
$AB \to BA$ (Note- We allow two nonterminals on the LHS.)
$AC \to CA$
$BC \to CB$
$BA \to AB$
$CA \to AC$
$CB \to BC$
$A \to a$
$B \to b$
$C \to c$
1) What lang does this generate?
$\{w : \#_a(w) = \#_b(w) = \#_c(w)\}$ which is NOT a CFL.
2) Context-Free means can replace (say) A by (say) α without looking at the **context** of A.

Examples of Context Sensitive Grammars

$S \to ABCS \mid e$
$AB \to BA$ (Note- We allow two nonterminals on the LHS.)
$AC \to CA$
$BC \to CB$
$BA \to AB$
$CA \to AC$
$CB \to BC$
$A \to a$
$B \to b$
$C \to c$

1) What lang does this generate?
$\{w : \#_a(w) = \#_b(w) = \#_c(w)\}$ which is NOT a CFL.
2) Context-Free means can replace (say) A by (say) α without looking at the **context** of A.
3) Context-Sensitive means can replace (say) A by (say) α AND look at what is around A. We actually allow more than that.

Context Sensitive Grammar for $\{a^n b^n c^n : n \in \mathbb{N}\}$

(**Warning** I came up with this so its intuitive but might be wrong.)

Context Sensitive Grammar for $\{a^n b^n c^n : n \in \mathbb{N}\}$

(**Warning** I came up with this so its intuitive but might be wrong.)
$S \rightarrow \$ABCX\$$

Context Sensitive Grammar for $\{a^n b^n c^n : n \in \mathbb{N}\}$

(**Warning** I came up with this so its intuitive but might be wrong.)
$S \rightarrow \$ABCX\$$
$X \rightarrow ABCX \mid e$

Context Sensitive Grammar for $\{a^n b^n c^n : n \in \mathbf{N}\}$

(**Warning** I came up with this so its intuitive but might be wrong.)
$S \rightarrow \$ABCX\$$
$X \rightarrow ABCX \mid e$
So now have $\$ABCABCABCABC \cdots ABC\$$.

Context Sensitive Grammar for $\{a^n b^n c^n : n \in \mathbf{N}\}$

(**Warning** I came up with this so its intuitive but might be wrong.)
$S \rightarrow \$ABCX\$$
$X \rightarrow ABCX \mid e$
So now have $\$ABCABCABCABC \cdots ABC\$$.

The next three rules get the A's to the left and the C's to the right:

Context Sensitive Grammar for $\{a^n b^n c^n : n \in \mathbf{N}\}$

(**Warning** I came up with this so its intuitive but might be wrong.)
$S \rightarrow \$ABCX\$$
$X \rightarrow ABCX \mid e$
So now have $\$ABCABCABCABC \cdots ABC\$$.

The next three rules get the A's to the left and the C's to the right:
$BA \rightarrow AB$

Context Sensitive Grammar for $\{a^n b^n c^n : n \in \mathbb{N}\}$

(**Warning** I came up with this so its intuitive but might be wrong.)
$S \rightarrow \$ABCX\$$
$X \rightarrow ABCX \mid e$
So now have $\$ABCABCABCABC \cdots ABC\$$.

The next three rules get the A's to the left and the C's to the right:
$BA \rightarrow AB$
$CA \rightarrow AC$

Context Sensitive Grammar for $\{a^n b^n c^n : n \in \mathbb{N}\}$

(**Warning** I came up with this so its intuitive but might be wrong.)
$S \rightarrow \$ABCX\$$
$X \rightarrow ABCX \mid e$
So now have $\$ABCABCABCABC \cdots ABC\$$.

The next three rules get the A's to the left and the C's to the right:
$BA \rightarrow AB$
$CA \rightarrow AC$
$CB \rightarrow BC$

Context Sensitive Grammar for $\{a^n b^n c^n : n \in \mathbf{N}\}$

(**Warning** I came up with this so its intuitive but might be wrong.)
$S \rightarrow \$ABCX\$$
$X \rightarrow ABCX \mid e$
So now have $\$ABCABCABCABC \cdots ABC\$$.

The next three rules get the A's to the left and the C's to the right:
$BA \rightarrow AB$
$CA \rightarrow AC$
$CB \rightarrow BC$
Can generate $\$A^n B^n C^n\$$ but can also generate other strings.
Want to replace A with a, etc, but only if of form $A^n B^n C^n$.

Context Sensitive Grammar for $\{a^n b^n c^n : n \in \mathbb{N}\}$

(**Warning** I came up with this so its intuitive but might be wrong.)
$S \rightarrow \$ABCX\$$
$X \rightarrow ABCX \mid e$
So now have $\$ABCABCABCABC \cdots ABC\$$.

The next three rules get the A's to the left and the C's to the right:
$BA \rightarrow AB$
$CA \rightarrow AC$
$CB \rightarrow BC$
Can generate $\$A^n B^n C^n\$$ but can also generate other strings.
Want to replace A with a, etc, but only if of form $A^n B^n C^n$.

$\$A \rightarrow a\$$

Context Sensitive Grammar for $\{a^n b^n c^n : n \in \mathbf{N}\}$

(**Warning** I came up with this so its intuitive but might be wrong.)
$S \rightarrow \$ABCX\$$
$X \rightarrow ABCX \mid e$
So now have $\$ABCABCABCABC \cdots ABC\$$.

The next three rules get the A's to the left and the C's to the right:
$BA \rightarrow AB$
$CA \rightarrow AC$
$CB \rightarrow BC$
Can generate $\$A^n B^n C^n\$$ but can also generate other strings.
Want to replace A with a, etc, but only if of form $A^n B^n C^n$.

$\$A \rightarrow a\$$
$C\$ \rightarrow \$c$

Context Sensitive Grammar for $\{a^n b^n c^n : n \in \mathbf{N}\}$

(**Warning** I came up with this so its intuitive but might be wrong.)

$S \rightarrow \$ABCX\$$

$X \rightarrow ABCX \mid e$

So now have $\$ABCABCABCABC \cdots ABC\$$.

The next three rules get the A's to the left and the C's to the right:

$BA \rightarrow AB$

$CA \rightarrow AC$

$CB \rightarrow BC$

Can generate $\$A^n B^n C^n\$$ but can also generate other strings.

Want to replace A with a, etc, but only if of form $A^n B^n C^n$.

$\$A \rightarrow a\$$

$C\$ \rightarrow \$c$

$\$ \rightarrow e$

Context Sensitive Grammar for $\{a^n b^n c^n : n \in \mathbf{N}\}$

(**Warning** I came up with this so its intuitive but might be wrong.)

$S \rightarrow \$ABCX\$$

$X \rightarrow ABCX \mid e$

So now have $\$ABCABCABCABC \cdots ABC\$$.

The next three rules get the A's to the left and the C's to the right:

$BA \rightarrow AB$

$CA \rightarrow AC$

$CB \rightarrow BC$

Can generate $\$A^n B^n C^n\$$ but can also generate other strings.

Want to replace A with a, etc, but only if of form $A^n B^n C^n$.

$\$A \rightarrow a\$$

$C\$ \rightarrow \$c$

$\$ \rightarrow e$

Grammar gen $\{a^n b^n c^n\}$. Need that it doesn't gen anything else.

Context Sensitive Grammar for $\{a^n b^n c^n : n \in \mathbb{N}\}$

(**Warning** I came up with this so its intuitive but might be wrong.)
$S \rightarrow \$ABCX\$$
$X \rightarrow ABCX \mid e$
So now have $\$ABCABCABCABC \cdots ABC\$$.

The next three rules get the A's to the left and the C's to the right:
$BA \rightarrow AB$
$CA \rightarrow AC$
$CB \rightarrow BC$
Can generate $\$A^nB^nC^n\$$ but can also generate other strings.
Want to replace A with a, etc, but only if of form $A^nB^nC^n$.

$\$A \rightarrow a\$$
$C\$ \rightarrow \$c$
$\$ \rightarrow e$

Grammar gen $\{a^nb^nc^n\}$. Need that it doesn't gen anything else.
Don't know so won't prove. Don't care so no extra credit for it.

Context Sensitive Grammar for $\{a^{n^2} : n \in N\}$

I **knew** that $\{a^{n^2} : n \in N\}$ is a CSL (will say why later).

Context Sensitive Grammar for $\{a^{n^2} : n \in N\}$

I **knew** that $\{a^{n^2} : n \in N\}$ is a CSL (will say why later).

I searched through books and the web to find a CSG for it since **students are our future!** and you **deserve** to know this grammar! (I sincerely meant this as a positive thing.)

Context Sensitive Grammar for $\{a^{n^2} : n \in \mathbb{N}\}$

I **knew** that $\{a^{n^2} : n \in \mathbb{N}\}$ is a CSL (will say why later).

I searched through books and the web to find a CSG for it since **students are our future!** and you **deserve** to know this grammar! (I sincerely meant this as a positive thing.)

I found an entry on Computer Science Stack Exchange so I do not know if it is correct, though I suspect it is.
Link: `https://cs.stackexchange.com/questions/53530/grammar-for-square-numbers-in-unary`

Context Sensitive Grammar for $\{a^{n^2} : n \in N\}$

I **knew** that $\{a^{n^2} : n \in N\}$ is a CSL (will say why later).

I searched through books and the web to find a CSG for it since **students are our future!** and you **deserve** to know this grammar! (I sincerely meant this as a positive thing.)

I found an entry on Computer Science Stack Exchange so I do not know if it is correct, though I suspect it is.
Link: `https://cs.stackexchange.com/questions/53530/grammar-for-square-numbers-in-unary`
In case that link goes away (plausible) and you are really eager to see the CSL (less plausible) next slide has the CSG for it (not quite).

Context Sensitive Grammar for $\{a^{n^2} : n \in \mathbf{N}\}$

$FD \rightarrow DG$
$AD \rightarrow DaA$
$aD \rightarrow Da$
$Aa \rightarrow aA$
$BD \rightarrow BH$
$Ha \rightarrow aH$
$HA \rightarrow AI$
$IA \rightarrow AI$
$IG \rightarrow AAF$
$FE \rightarrow E$
$B \rightarrow e$
$AE \rightarrow E$
$E \rightarrow e$
(Last four rules not allowed in a CSG but this can be dealt with.)

Definition of Context Sensitive Grammars

Def A **Context Sensitive Grammar** is a tuple $G = (N, \Sigma, R, S)$

Definition of Context Sensitive Grammars

Def A **Context Sensitive Grammar** is a tuple $G = (N, \Sigma, R, S)$

- N is a finite set of **nonterminals**.

Definition of Context Sensitive Grammars

Def A **Context Sensitive Grammar** is a tuple $G = (N, \Sigma, R, S)$

- N is a finite set of **nonterminals**.
- Σ is a finite **alphabet**. Note $\Sigma \cap N = \emptyset$.

Definition of Context Sensitive Grammars

Def A **Context Sensitive Grammar** is a tuple $G = (N, \Sigma, R, S)$

- N is a finite set of **nonterminals**.
- Σ is a finite **alphabet**. Note $\Sigma \cap N = \emptyset$.
- $R \subseteq (N \cup \Sigma)^* N (N \cup \Sigma)^* \times (N \cup \Sigma)^*$ and are called **Rules**.

Definition of Context Sensitive Grammars

Def A **Context Sensitive Grammar** is a tuple $G = (N, \Sigma, R, S)$

- N is a finite set of **nonterminals**.
- Σ is a finite **alphabet**. Note $\Sigma \cap N = \emptyset$.
- $R \subseteq (N \cup \Sigma)^* N (N \cup \Sigma)^* \times (N \cup \Sigma)^*$ and are called **Rules**.
- **Important** For every rule the $\alpha \to \beta$, $|\alpha| \leq |\beta|$. (CSG's are given that violate this but can be put into this form.)

Definition of Context Sensitive Grammars

Def A **Context Sensitive Grammar** is a tuple $G = (N, \Sigma, R, S)$

- N is a finite set of **nonterminals**.
- Σ is a finite **alphabet**. Note $\Sigma \cap N = \emptyset$.
- $R \subseteq (N \cup \Sigma)^* N (N \cup \Sigma)^* \times (N \cup \Sigma)^*$ and are called **Rules**.
- **Important** For every rule the $\alpha \to \beta$, $|\alpha| \le |\beta|$. (CSG's are given that violate this but can be put into this form.)
- $S \in N$, the **start symbol**.

Definition of Context Sensitive Grammars

Def A **Context Sensitive Grammar** is a tuple $G = (N, \Sigma, R, S)$

- N is a finite set of **nonterminals**.
- Σ is a finite **alphabet**. Note $\Sigma \cap N = \emptyset$.
- $R \subseteq (N \cup \Sigma)^* N (N \cup \Sigma)^* \times (N \cup \Sigma)^*$ and are called **Rules**.
- **Important** For every rule the $\alpha \rightarrow \beta$, $|\alpha| \leq |\beta|$. (CSG's are given that violate this but can be put into this form.)
- $S \in N$, the **start symbol**.

Note

Definition of Context Sensitive Grammars

Def A **Context Sensitive Grammar** is a tuple $G = (N, \Sigma, R, S)$

- N is a finite set of **nonterminals**.
- Σ is a finite **alphabet**. Note $\Sigma \cap N = \emptyset$.
- $R \subseteq (N \cup \Sigma)^* N (N \cup \Sigma)^* \times (N \cup \Sigma)^*$ and are called **Rules**.
- **Important** For every rule the $\alpha \rightarrow \beta$, $|\alpha| \le |\beta|$. (CSG's are given that violate this but can be put into this form.)
- $S \in N$, the **start symbol**.

Note

1) The LHS must have at least one nonterminal.

Definition of Context Sensitive Grammars

Def A **Context Sensitive Grammar** is a tuple $G = (N, \Sigma, R, S)$

- N is a finite set of **nonterminals**.
- Σ is a finite **alphabet**. Note $\Sigma \cap N = \emptyset$.
- $R \subseteq (N \cup \Sigma)^* N (N \cup \Sigma)^* \times (N \cup \Sigma)^*$ and are called **Rules**.
- **Important** For every rule the $\alpha \to \beta$, $|\alpha| \le |\beta|$. (CSG's are given that violate this but can be put into this form.)
- $S \in N$, the **start symbol**.

Note
1) The LHS must have at least one nonterminal.
2) There are alternative definitions that are equivalent, which I won't get into here.

L(G)

If A is non-terminal then the CSG gives us gives us rules like:

- $A \to AB$
- $A \to a$

L(G)

If A is non-terminal then the CSG gives us gives us rules like:

- $A \rightarrow AB$
- $A \rightarrow a$

$A \Rightarrow AAaASdD$ is valid when the RHS is a string of **terminals and non-terminals** that can be produced from A (LHS is a single **non-terminal**).

L(G)

If A is non-terminal then the CSG gives us gives us rules like:

- $A \rightarrow AB$
- $A \rightarrow a$

$A \Rightarrow AAaASdD$ is valid when the RHS is a string of **terminals and non-terminals** that can be produced from A (LHS is a single **non-terminal**).

Examples: If we have rules

$A \rightarrow BCD$

$BC \rightarrow BcA$

$BcA \rightarrow Aa$

...

Then we have

$A \rightarrow BCD \rightarrow BcAD \rightarrow AaD$

So $A \Rightarrow AaD$

L(G)

If A is non-terminal then the CSG gives us gives us rules like:

- $A \rightarrow AB$
- $A \rightarrow a$

$A \Rightarrow AAaASdD$ is valid when the RHS is a string of **terminals and non-terminals** that can be produced from A (LHS is a single **non-terminal**).

Examples: If we have rules

$A \rightarrow BCD$

$BC \rightarrow BcA$

$BcA \rightarrow Aa$

...

Then we have

$A \rightarrow BCD \rightarrow BcAD \rightarrow AaD$

So $A \Rightarrow AaD$

Then, if w is string of **non-terminals only**, we define $L(G)$ by:

L(G)

If A is non-terminal then the CSG gives us gives us rules like:

- $A \to AB$
- $A \to a$

$A \Rightarrow AAaASdD$ is valid when the RHS is a string of **terminals and non-terminals** that can be produced from A (LHS is a single **non-terminal**).

Examples: If we have rules
$A \to BCD$
$BC \to BcA$
$BcA \to Aa$
...
Then we have
$A \to BCD \to BcAD \to AaD$
So $A \Rightarrow AaD$
Then, if w is string of **non-terminals only**, we define $L(G)$ by:

$$L(G) = \{w \in \Sigma^* \mid S \Rightarrow w\}$$

Example of a Lang that is NOT a CSL

We'll come back to this later.

CLOSURE PROPERTIES

Closure Properties

If L_1, L_2 are Context Sensitive Languages then

1. Is $L_1 \cup L_2$ a context sensitive Lang?

Closure Properties

If L_1, L_2 are Context Sensitive Languages then

1. Is $L_1 \cup L_2$ a context sensitive Lang? Yes, Easy.

Closure Properties

If L_1, L_2 are Context Sensitive Languages then

1. Is $L_1 \cup L_2$ a context sensitive Lang? Yes, Easy.
2. Is $L_1 \cap L_2$ a context sensitive Lang?

Closure Properties

If L_1, L_2 are Context Sensitive Languages then

1. Is $L_1 \cup L_2$ a context sensitive Lang? Yes, Easy.
2. Is $L_1 \cap L_2$ a context sensitive Lang? Yes, Hard.

Closure Properties

If L_1, L_2 are Context Sensitive Languages then

1. Is $L_1 \cup L_2$ a context sensitive Lang? Yes, Easy.
2. Is $L_1 \cap L_2$ a context sensitive Lang? Yes, Hard.
3. Is $L_1 \cdot L_2$ a context sensitive Lang?

Closure Properties

If L_1, L_2 are Context Sensitive Languages then

1. Is $L_1 \cup L_2$ a context sensitive Lang? Yes, Easy.
2. Is $L_1 \cap L_2$ a context sensitive Lang? Yes, Hard.
3. Is $L_1 \cdot L_2$ a context sensitive Lang? Yes, Easy.

Closure Properties

If L_1, L_2 are Context Sensitive Languages then

1. Is $L_1 \cup L_2$ a context sensitive Lang? Yes, Easy.
2. Is $L_1 \cap L_2$ a context sensitive Lang? Yes, Hard.
3. Is $L_1 \cdot L_2$ a context sensitive Lang? Yes, Easy.
4. Is $\overline{L_1}$ a context sensitive Lang?

Closure Properties

If L_1, L_2 are Context Sensitive Languages then

1. Is $L_1 \cup L_2$ a context sensitive Lang? Yes, Easy.
2. Is $L_1 \cap L_2$ a context sensitive Lang? Yes, Hard.
3. Is $L_1 \cdot L_2$ a context sensitive Lang? Yes, Easy.
4. Is $\overline{L_1}$ a context sensitive Lang? Yes, Hard.

Closure Properties

If L_1, L_2 are Context Sensitive Languages then

1. Is $L_1 \cup L_2$ a context sensitive Lang? Yes, Easy.
2. Is $L_1 \cap L_2$ a context sensitive Lang? Yes, Hard.
3. Is $L_1 \cdot L_2$ a context sensitive Lang? Yes, Easy.
4. Is $\overline{L_1}$ a context sensitive Lang? Yes, Hard.
5. IS L_1^* is a context sensitive Lang?

Closure Properties

If L_1, L_2 are Context Sensitive Languages then

1. Is $L_1 \cup L_2$ a context sensitive Lang? Yes, Easy.
2. Is $L_1 \cap L_2$ a context sensitive Lang? Yes, Hard.
3. Is $L_1 \cdot L_2$ a context sensitive Lang? Yes, Easy.
4. Is $\overline{L_1}$ a context sensitive Lang? Yes, Hard.
5. IS L_1^* is a context sensitive Lang? Yes, Easy.

MISC

MISC

1) There is a no pumping theorem for CSL's.

MISC

1) There is a no pumping theorem for CSL's.
2) Recall:

MISC

1) There is a no pumping theorem for CSL's.
2) Recall:
DFA's are **Recognizers**, Regex are **Generators**.

MISC

1) There is a no pumping theorem for CSL's.
2) Recall:
DFA's are **Recognizers**, Regex are **Generators**.
PDA's are **Recognizers**, CFG's are **Generators**.

MISC

1) There is a no pumping theorem for CSL's.
2) Recall:
DFA's are **Recognizers**, Regex are **Generators**.
PDA's are **Recognizers**, CFG's are **Generators**.
CSL's are **Generators**.

MISC

1) There is a no pumping theorem for CSL's.
2) Recall:
DFA's are **Recognizers**, Regex are **Generators**.
PDA's are **Recognizers**, CFG's are **Generators**.
CSL's are **Generators**.
There is a **Recognizer** equivalent to it:

LBA's

MISC

1) There is a no pumping theorem for CSL's.
2) Recall:
DFA's are **Recognizers**, Regex are **Generators**.
PDA's are **Recognizers**, CFG's are **Generators**.
CSL's are **Generators**.
There is a **Recognizer** equivalent to it:

LBA's

LBA stands for **Linear Bounded Automata**.
They are nondeterministic Turing machines with $O(n)$ space.

MISC

1) There is a no pumping theorem for CSL's.
2) Recall:
DFA's are **Recognizers**, Regex are **Generators**.
PDA's are **Recognizers**, CFG's are **Generators**.
CSL's are **Generators**.
There is a **Recognizer** equivalent to it:

LBA's

LBA stands for **Linear Bounded Automata**.
They are nondeterministic Turing machines with $O(n)$ space.
The proof that LBA-recognizers and CSG-generators are equivalent is messy so we won't be doing it. We won't deal with LBA's in this course at all.

Later is Now

I said earlier:

I **knew** *that* $\{a^{n^2} : n \in \mathsf{N}\}$ *is a CSL (will say why later).*

Later is Now

I said earlier:

I **knew** *that* $\{a^{n^2} : n \in N\}$ *is a CSL (will say why later).*

It is easy to write an LBA for $\{a^{n^2} : n \in N\}$

Later is Now

I said earlier:

I **knew** *that* $\{a^{n^2} : n \in \mathbb{N}\}$ *is a CSL (will say why later).*

It is easy to write an LBA for $\{a^{n^2} : n \in \mathbb{N}\}$
Hence it is easy to show that $\{a^{n^2} : n \in \mathbb{N}\}$ and many other languages are CSL's without using CSG's.

Example of a Lang that is NOT a CSL?

In this slide we only refer to decidable languages.

Example of a Lang that is NOT a CSL?

In this slide we only refer to decidable languages.
1) Not known how CSL and NP compare. Likely incomparable.

Example of a Lang that is NOT a CSL?

In this slide we only refer to decidable languages.
1) Not known how CSL and NP compare. Likely incomparable.
2) Proving incompatibility seems hard.

Example of a Lang that is NOT a CSL?

In this slide we only refer to decidable languages.
1) Not known how CSL and NP compare. Likely incomparable.
2) Proving incompatibility seems hard.
(Proving hard to show inc was part of my PhD Thesis.)

Example of a Lang that is NOT a CSL?

In this slide we only refer to decidable languages.
1) Not known how CSL and NP compare. Likely incomparable.
2) Proving incompatibility seems hard.
(Proving hard to show inc was part of my PhD Thesis.)
3) Proving a langauge is NOT CSL seems hard.

Example of a Lang that is NOT a CSL?

In this slide we only refer to decidable languages.
1) Not known how CSL and NP compare. Likely incomparable.
2) Proving incompatibility seems hard.
(Proving hard to show inc was part of my PhD Thesis.)
3) Proving a langauge is NOT CSL seems hard.
4) One can construct a (contrived) language that is NOT CSL.

Example of a Lang that is NOT a CSL?

In this slide we only refer to decidable languages.
1) Not known how CSL and NP compare. Likely incomparable.
2) Proving incompatibility seems hard.
(Proving hard to show inc was part of my PhD Thesis.)
3) Proving a langauge is NOT CSL seems hard.
4) One can construct a (contrived) language that is NOT CSL.
5) I know of **one** natural language that is provably not CSL:

Example of a Lang that is NOT a CSL?

In this slide we only refer to decidable languages.

1) Not known how CSL and NP compare. Likely incomparable.

2) Proving incompatibility seems hard.

(Proving hard to show inc was part of my PhD Thesis.)

3) Proving a langauge is NOT CSL seems hard.

4) One can construct a (contrived) language that is NOT CSL.

5) I know of **one** natural language that is provably not CSL:

Given two **trex** α *and* β, *is* $L(\alpha) = L(\beta)$.

Example of a Lang that is NOT a CSL?

In this slide we only refer to decidable languages.
1) Not known how CSL and NP compare. Likely incomparable.
2) Proving incompatibility seems hard.
(Proving hard to show inc was part of my PhD Thesis.)
3) Proving a langauge is NOT CSL seems hard.
4) One can construct a (contrived) language that is NOT CSL.
5) I know of **one** natural language that is provably not CSL:

Given two **trex** α *and* β, *is* $L(\alpha) = L(\beta)$.

Open question Some variants of Chess and Go **might be** provably not CSL.

Comparing Reg, CFL, CSL

We have a table of Reg, CFL, CSL.
Y is YES. N is NO
E is Easy. H is Hard.

Comparing Reg, CFL, CSL

We have a table of Reg, CFL, CSL.
Y is YES. N is NO
E is Easy. H is Hard.

Lang	Rcg	Gen	$\cup$	$\cap$	$\cdot$	*	comp	PL
Reg	DFA	Rgx	Y-E	Y-E	Y-E	Y-E	Y-E	Y
CFG	PDA	CFG	Y-E	N-E	Y-E	Y-E	N-E	Y
CSG	LBA	CSG	Y-E	Y-H	Y-E	Y-E	Y-E	N

1. Proving sets are not Regular is **Easy**.
2. Proving sets are not Context-Free is **Easy**.
3. Proving sets are not Context-Sensitive is **Hard**.