Foundations of Computing

Lecture 11

Arkady Yerukhimovich

February 21, 2023

Outline

1. Lecture 10 Review
2. The CFL Pumping Lemma
3. Midterm Review

Lecture 10 Review

- CFG == PDA
  - Construct PDA from CFG
  - Construct CFG from PDA
- CFG Pumping Lemma

Lecture 10 Review

- CFG == PDA
  - Construct PDA from CFG
  - Construct CFG from PDA
- CFG Pumping Lemma

Today

- How to use the CFG pumping lemma
- Midterm review

Outline

The CFL Pumping Lemma

Theorem

If L is a CFL, then there exists a pumping length p s.t. for any $s \in L$, with $|s| \geq p$, s can be divided into 5 pieces $s = uvxyz$ satisfying:

1. For each $i \geq 0$, $uv^ixy^iz \in L$
2. $|vy| > 0$
3. $|vxy| \leq p$

The CFL Pumping Lemma

Theorem

If L is a CFL, then there exists a pumping length p s.t. for any $s \in L$, with $|s| \geq p$, s can be divided into 5 pieces $s = uvxyz$ satisfying:

1. For each $i \geq 0$, $uv^ixy^iz \in L$
2. $|vy| > 0$
3. $|vxy| \leq p$

Pumping lemma in math notation:
$\exists p$ s.t $\forall s \in L, |s| \geq p$, $\exists$ partition $s = uvxyz$ s.t. $\forall i, uv^ixy^iz \in L$

The CFL Pumping Lemma

Theorem

If L is a CFL, then there exists a pumping length p s.t. for any $s \in L$, with $|s| \geq p$, s can be divided into 5 pieces $s = uvxyz$ satisfying:

1. For each $i \geq 0$, $uv^i xy^i z \in L$
2. $|vy| > 0$
3. $|vxy| \leq p$

Pumping lemma in math notation:
$\exists p$ s.t $\forall s \in L, |s| \geq p$, $\exists$ partition $s = uvxyz$ s.t. $\forall i, uv^i xy^i z \in L$

Negation of pumping lemma:
$\forall p$, $\exists s \in L, |s| \geq p$ s.t. $\forall$ partitions $s = uvxyz$ $\exists$ i s.t. $uv^i xy^i z \notin L$

Using the CFL Pumping Lemma

We use the CFL pumping lemma to prove that L is not a CFL similarly to how we used the regular language pumping lemma.

Using the CFL Pumping Lemma

We use the CFL pumping lemma to prove that L is not a CFL similarly to how we used the regular language pumping lemma.

Specifically:

- Consider the negation:

$$\forall p,\ \exists s \in L, |s| \geq p \text{ s.t. } \forall \text{ partitions } s = uvxyz\ \exists i \text{ s.t. } uv^i xy^i z \notin L$$

Using the CFL Pumping Lemma

We use the CFL pumping lemma to prove that L is not a CFL similarly to how we used the regular language pumping lemma.

Specifically:

- Consider the negation:

$$\forall p,\ \exists s \in L, |s| \geq p \text{ s.t. } \forall \text{ partitions } s = uvxyz\ \exists i \text{ s.t. } uv^ixy^iz \notin L$$

- So, we need to find such an s and prove that for any way to partition it, it cannot be pumped

The Proof Procedure

To use the pumping lemma to prove that L is not CFL, we do the following:

The Proof Procedure

To use the pumping lemma to prove that L is not CFL, we do the following:

1. Assume that L is CFL

The Proof Procedure

To use the pumping lemma to prove that L is not CFL, we do the following:

1. Assume that L is CFL
2. Use pumping lemma to guarantee pumping length p, s.t. all s with $|s| \geq p$ can be pumped

The Proof Procedure

To use the pumping lemma to prove that L is not CFL, we do the following:

1. Assume that L is CFL
2. Use pumping lemma to guarantee pumping length p, s.t. all s with $|s| > p$ can be pumped
3. Pick some $s \in L$ with $|s| \geq p$

The Proof Procedure

To use the pumping lemma to prove that L is not CFL, we do the following:

1. Assume that L is CFL
2. Use pumping lemma to guarantee pumping length p, s.t. all s with $|s| > p$ can be pumped
3. Pick some $s \in L$ with $|s| \geq p$
4. Demonstrate that s cannot be pumped
   - For each possible division $\cancel{w}\ s = uvxyz$ (with $|vy| > 0$ and $|vxy| \leq p$), find an integer i such that $uv^ixy^iz \notin L$

The Proof Procedure

To use the pumping lemma to prove that L is not CFL, we do the following:

1. Assume that L is CFL
2. Use pumping lemma to guarantee pumping length p, s.t. all s with $|s| > p$ can be pumped
3. Pick some $s \in L$ with $|s| \geq p$
4. Demonstrate that s cannot be pumped
    - For each possible division $w = uvxyz$ (with $|vy| > 0$ and $|vxy| \leq p$), find an integer i such that $uv^ixy^iz \notin L$
5. Contradiction!!!

Example 1

Consider $L = \{a^n b^n c^n \mid n \geq 0\}$, prove L is not CFL

Example 1

Consider $L = \{a^n b^n c^n \mid n \geq 0\}$, prove L is not CFL

Proof:

1. Assume L is CFL, and let p be the pumping length

Example 1

Consider $L = \{a^n b^n c^n \mid n \geq 0\}$, prove L is not CFL

Proof:

1. Assume L is CFL, and let p be the pumping length
2. Choose $s = a^p b^p c^p \in L$

Example 1

Consider $L = \{a^n b^n c^n \mid n \geq 0\}$, prove L is not CFL

Proof:

1. Assume L is CFL, and let p be the pumping length
2. Choose $s = a^p b^p c^p \in L$
3. By pumping lemma, $s = uvxyz$ s.t. $uv^i xy^i z \in L$ for all i
4. Complete proof by considering all possible values for v, y

Example 1

Consider $L = \{a^n b^n c^n \mid n \geq 0\}$, prove L is not CFL

Proof:

1. Assume L is CFL, and let p be the pumping length
2. Choose $s = a^p b^p c^p \in L$
3. By pumping lemma, $s = uvxyz$ s.t. $uv^i xy^i z \in L$ for all i
4. Complete proof by considering all possible values for v, y
    - v and y both have only one type of symbol (e.g., $v = a^\ell$ and $y = b^{\ell'}$) then $uv^i xy^i z$ has more a's and b's than c's, so is not in L

Example 1

Consider $L = \{a^n b^n c^n \mid n \geq 0\}$, prove L is not CFL

Proof:

1. Assume L is CFL, and let p be the pumping length
2. Choose $s = a^p b^p c^p \in L$
3. By pumping lemma, $s = uvxyz$ s.t. $uv^i xy^i z \in L$ for all i
4. Complete proof by considering all possible values for v, y
    - v and y both have only one type of symbol (e.g., $v = a^\ell$ and $y = b^{\ell'}$) then $uv^i xy^i z$ has more a's and b's than c's, so is not in L
    - If either v or y have more than one type of symbol, $uv^i xy^i z$ will have alternating symbols, so not in L

$i=2$

$s = \underbrace{a}_{u}\underbrace{ab}_{v}\underbrace{b}_{x}\underbrace{c}_{y}\underbrace{c}_{z} \rightarrow a\,\underline{abab}\,b\,ccc$

Example 1

Consider $L = \{a^n b^n c^n \mid n \geq 0\}$, prove L is not CFL

Proof:

1. Assume L is CFL, and let p be the pumping length
2. Choose $s = a^p b^p c^p \in L$
3. By pumping lemma, $s = uvxyz$ s.t. $uv^i xy^i z \in L$ for all i
4. Complete proof by considering all possible values for v, y
   - v and y both have only one type of symbol (e.g., $v = a^\ell$ and $y = b^{\ell'}$) then $uv^i xy^i z$ has more a's and b's than c's, so is not in L
   - If either v or y have more than one type of symbol, $uv^i xy^i z$ will have alternating symbols, so not in L
5. Contradiction – Hence L is not CFL

Example 2

Consider $L = \{ww \mid w \in \{0,1\}^*\}$, prove L is not CFL

$L = \{ww^R \in CFL\}$

Example 2

Consider $L = \{ww \mid w \in \{0,1\}^*\}$, prove L is not CFL

Proof:

1. Assume L is CFL, and let p be the pumping length
2. Try 1: Choose $s = 0^p10^p1 \in L$

$s = uvxyz$

Example 2

Consider $L = \{ww \mid w \in \{0,1\}^*\}$, prove L is not CFL

Proof:

1. Assume L is CFL, and let p be the pumping length
2. Try 1: Choose $s = 0^p 1 0^p 1 \in L$
3. Try 2: Choose $s = 0^p 1^p 0^p 1^p \in L$

Example 2

Consider $L = \{ww \mid w \in \{0,1\}^*\}$, prove L is not CFL

Proof:

1. Assume L is CFL, and let p be the pumping length
2. Try 1: Choose $s = 0^p 1 0^p 1 \in L$
3. Try 2: Choose $s = 0^p 1^p 0^p 1^p \in L$
4. Consider all possible cases for vxy ($|vxy| \leq p$)

$s = u\underline{vxy}z$

Example 2

Consider $L = \{ww \mid w \in \{0,1\}^*\}$, prove L is not CFL

Proof:

1. Assume L is CFL, and let p be the pumping length
2. Try 1: Choose $s = 0^p10^p1 \in L$
3. Try 2: Choose $s = 0^p1^p|0^p1^p \in L$
4. Consider all possible cases for vxy ($|vxy| \leq p$)
   - vxy does not contain the midpoint of s
     - vxy is left of center – pumping moves a 1 into first character of right half

000 111 | 000 111

00 00 111|1 000 111

Example 2

Consider $L = \{ww \mid w \in \{0,1\}^*\}$, prove L is not CFL

Proof:

1. Assume L is CFL, and let p be the pumping length
2. Try 1: Choose $s = 0^p10^p1 \in L$
3. Try 2: Choose $s = 0^p1^p0^p1^p \in L$
4. Consider all possible cases for vxy ($|vxy| \leq p$)
   - vxy does not contain the midpoint of s
     - vxy is left of center – pumping moves a 1 into first character of right half
     - vxy is left of center – pumping moves a 0 into last character of left half

Example 2

Consider $L = \{ww \mid w \in \{0,1\}^*\}$, prove L is not CFL

Proof:

1. Assume L is CFL, and let p be the pumping length
2. Try 1: Choose $s = 0^p 1 0^p 1 \in L$
3. Try 2: Choose $s = 0^p 1^p 0^p 1^p \in L$
4. Consider all possible cases for vxy ($|vxy| \le p$)
   - vxy does not contain the midpoint of s
     - vxy is left of center – pumping moves a 1 into first character of right half
     - vxy is left of center – pumping moves a 0 into last character of left half
   - vxy does contain the midpoint of s – pumping makes this not match unpumped parts

Example 2

Consider $L = \{ww \mid w \in \{0,1\}^*\}$, prove L is not CFL

Proof:

1. Assume L is CFL, and let p be the pumping length
2. Try 1: Choose $s = 0^p10^p1 \in L$
3. Try 2: Choose $s = 0^p1^p0^p1^p \in L$
4. Consider all possible cases for vxy ($|vxy| \leq p$)
   - vxy does not contain the midpoint of s
     - vxy is left of center – pumping moves a 1 into first character of right half
     - vxy is left of center – pumping moves a 0 into last character of left half
   - vxy does contain the midpoint of s – pumping makes this not match unpumped parts
5. Contradiction – Hence L is not CFL

Outline

Topics

1. DFA
   - Know what it means for a DFA to accept a string
   - Know what it means for DFA to accept/recognize a language

Topics

1. DFA
    - Know what it means for a DFA to accept a string
    - Know what it means for DFA to accept/recognize a language
2. Building DFA
    - Given description of L build a DFA for it

Topics

1. DFA
    - Know what it means for a DFA to accept a string
    - Know what it means for DFA to accept/recognize a language
2. Building DFA
    - Given description of L build a DFA for it
3. Regular Languages
    - Know what they are
    - Recall closure properties of regular languages (complement, union, intersection, concatenation, $*$ closure)

Topics

1. DFA
   - Know what it means for a DFA to accept a string
   - Know what it means for DFA to accept/recognize a language
2. Building DFA
   - Given description of L build a DFA for it
3. Regular Languages
   - Know what they are
   - Recall closure properties of regular languages (complement, union, intersection, concatenation, $*$ closure)
4. NFAs

Topics

1. DFA
   - Know what it means for a DFA to accept a string
   - Know what it means for DFA to accept/recognize a language
2. Building DFA
   - Given description of L build a DFA for it
3. Regular Languages
   - Know what they are
   - Recall closure properties of regular languages (complement, union, intersection, concatenation, $*$ closure)
4. NFAs
   - Know when an NFA accepts a string/language
   - Know when it doesn't accept a string/language

Topics

1. DFA
   - Know what it means for a DFA to accept a string
   - Know what it means for DFA to accept/recognize a language
2. Building DFA
   - Given description of L build a DFA for it
3. Regular Languages
   - Know what they are
   - Recall closure properties of regular languages (complement, union, intersection, concatenation, $*$ closure)
4. NFAs
   - Know when an NFA accepts a string/language
   - Know when it doesn't accept a string/language
   - Remember the definition of non-determinism – only need there to exist a part to accept

Topics

1. DFA
   - Know what it means for a DFA to accept a string
   - Know what it means for DFA to accept/recognize a language
2. Building DFA
   - Given description of L build a DFA for it
3. Regular Languages
   - Know what they are
   - Recall closure properties of regular languages (complement, union, intersection, concatenation, $*$ closure)
4. NFAs
   - Know when an NFA accepts a string/language
   - Know when it doesn't accept a string/language
   - Remember the definition of non-determinism – only need there to exist a part to accept
   - Be able to build an NFA from a language description

Topics

1. DFA
   - Know what it means for a DFA to accept a string
   - Know what it means for DFA to accept/recognize a language
2. Building DFA
   - Given description of L build a DFA for it
3. Regular Languages
   - Know what they are
   - Recall closure properties of regular languages (complement, union, intersection, concatenation, $*$ closure)
4. NFAs
   - Know when an NFA accepts a string/language
   - Know when it doesn't accept a string/language
   - Remember the definition of non-determinism – only need there to exist a part to accept
   - Be able to build an NFA from a language description
   - NFA to DFA using the finger method

Topics

5. Regular Expressions

Topics

5. Regular Expressions
   - Be able to build an RE for a language

Topics

5. Regular Expressions
    - Be able to build an RE for a language
    - RE to NFA
    - NFA to RE

Topics

5. Regular Expressions
    - Be able to build an RE for a language
    - RE to NFA
    - NFA to RE
6. Regular Language Pumping Lemma

Topics

5. Regular Expressions
   - Be able to build an RE for a language
   - RE to NFA
   - NFA to RE
6. Regular Language Pumping Lemma
   - Remember statement as sequence of quantifiers

Topics

5. Regular Expressions
   - Be able to build an RE for a language
   - RE to NFA
   - NFA to RE
6. Regular Language Pumping Lemma
   - Remember statement as sequence of quantifiers
   - Understand why it is true (state of NFA must repeat)

Topics

5. Regular Expressions
   - Be able to build an RE for a language
   - RE to NFA
   - NFA to RE
6. Regular Language Pumping Lemma
   - Remember statement as sequence of quantifiers
   - Understand why it is true (state of NFA must repeat)
   - Understand how to use it.

Topics

5. Regular Expressions
   - Be able to build an RE for a language
   - RE to NFA
   - NFA to RE
6. Regular Language Pumping Lemma
   - Remember statement as sequence of quantifiers
   - Understand why it is true (state of NFA must repeat)
   - Understand how to use it.
   - Also know how to prove languages not regular using closure properties

Topics

7. PDAs

Topics

7. PDAs
    - Know what strings/languages are accepted
    - Know what strings/languages are not

Topics

7. PDAs
    - Know what strings/languages are accepted
    - Know what strings/languages are not
    - Be able to build a PDA given a language description

Topics

7. PDAs
   - Know what strings/languages are accepted
   - Know what strings/languages are not
   - Be able to build a PDA given a language description
8. Context-free Grammars (CFG)

Topics

7. PDAs
   - Know what strings/languages are accepted
   - Know what strings/languages are not
   - Be able to build a PDA given a language description
8. Context-free Grammars (CFG)
   - Remember what this means

Topics

7. PDAs
    - Know what strings/languages are accepted
    - Know what strings/languages are not
    - Be able to build a PDA given a language description
8. Context-free Grammars (CFG)
    - Remember what this means
    - Be able to construct one from language description

Topics

7. PDAs
   - Know what strings/languages are accepted
   - Know what strings/languages are not
   - Be able to build a PDA given a language description
8. Context-free Grammars (CFG)
   - Remember what this means
   - Be able to construct one from language description
   - Remember what a derivation is and what a parse tree is

Topics

7. PDAs
   - Know what strings/languages are accepted
   - Know what strings/languages are not
   - Be able to build a PDA given a language description
8. Context-free Grammars (CFG)
   - Remember what this means
   - Be able to construct one from language description
   - Remember what a derivation is and what a parse tree is
   - PDA == CFG (at a high level)
9. CFL pumping lemma
   - There will not be any questions on the CFG pumping lemma on the exam
   - But, there will be on the next homework

Exam Format

- 7 questions – most have multiple parts
- Covers most of the material outlined above
- 2 questions requiring proofs, the rest are more constructive
- Some yes/no questions

Don't Forget

- Exam is in class on Thursday 11:10-12:25, don't be late!
- You can bring two 8.5 $\times$ 11 piece of paper ~~and wa~~

Any Questions?

$L = \{ w \overline{w}^R \mid w \in \{0,1\}^* \}$

0101

1100

110100

1. For every character in w push it on stack
2. Non-deterministically decide when w ends
3. Pop X, check if input $= \overline{X}$
4. If stack is empty when out of input, accept

Any Questions?

$L = \{ w \overline{w^R} \}$

$S \to 0 S 1$

$S \to 1 S 0$

$S \to \epsilon$

Any Questions?

$L = \{w \mid w$ has a substring $11\}$ $w \in \{0,1\}^*$

~~ta~~

Σ^*

$(0 \cup 1)^* 11 (0 \cup 1)^*$