CS420

Introduction to the Theory of Computation

Lecture: Module 2 recap

Tiago Cogumbreiro
Disclaimer: PDA semantics

- The PDAs we learned are slightly different than the ones in the book.
- The definition of PDAs we introduced in our lecture have two special operations (stack-empty? and clear-stack); in the book the semantics are a bit simpler, but the diagrams become a bit more verbose.
- This simplifies the design of PDAs.
- Our version is a super-set of the book (in terms of state-diagram), so the examples in the book should all have the same meaning.
- You can use either version in Mini-Test 2.
Mini-Test 2

- Written exam: 80 points (out of 100)
- Coq script: 35 points (out of 100)
- Total: 115 points (15 extra points max)

About

- 13 questions
- Max value per question: 15 points
- Took me 20 minutes to solve
Exercise 1

\[ A \rightarrow 0A1 | B \]
\[ B \rightarrow 1A | \epsilon \]

Exercise 3 of Lesson 10

Convert the following grammar into a PDA
Exercise 1

\[ A \rightarrow 0A1 \mid B \]
\[ B \rightarrow 1A \mid \epsilon \]

Exercise 3 of Lesson 10

Convert the following grammar into a PDA
Exercise 2

Given that $L_2 = \{0^i1^j0^k \mid i = j = k \lor k = i + j\}$ is not context-free, show that
\[ \{0^n1^n0^n \mid n \geq 0\} \] is not context-free without using the pumping lemma.
Given that $L_2 = \{0^i1^j0^k \mid i = j = k \lor k = i + j\}$ is not context-free, show that $\{0^n1^n0^n \mid n \geq 0\}$ is not context-free without using the pumping lemma.

1. $L_2 = A \cup B$ where $A = \{0^i1^j0^k \mid k = i + j\}$ and $B = \{0^n1^n0^n \mid n \geq 0\}$
2. We know that if $L$ CF and $L'$ CF, then $L \cup L'$ CF.
3. Applying the contra-positive we have that: $L \cup L'$ not CF implies that not $(L$ is CF and $L'$ CF)
4. Since $\neg(P \land Q) \implies \neg P \lor \neg Q$, thus either $A$ not CF or $B$ not CF.
5. We have that $B$ is CF (from HW5), hence $A$ not CF

These exercises always use contra-positive of the closure properties (union, star, concat)

You are asked to show that given $L$ is not context-free, then $L'$ is not context free:

1. Show that $L'$ can be obtained from $L$ using union/concat/star
2. Apply the contra-positive of the closure property
3. Conclude the goal
Exercise 3

Let be $L$ context-free. Is $L_3 = L^*$

- regular?
- not-regular?
- context-free?
- not-context-free?

If $L_3$ is context-free, prove it.
If $L_3$ is not-context-free, prove it.
Exercise 3

Let be $L$ context-free. Is $L_3 = L^*$

- regular?
- not-regular?
- context-free?
- not-context-free?

If $L_3$ is context-free, prove it. **Solution:** $L_3$ is context-free, because $L_3 = L^*$ and we know that if $L$ is context-free, then $L_3$ is context-free.

If $L_3$ is not-context-free, prove it.
Exercise 4

For any language $L_1$ there exists a regular language $L_2$ such that $L_1 \subseteq L_2$. 
Exercise 4

For any language $L_1$ there exists a regular language $L_2$ such that $L_1 \subseteq L_2$.

True.

Proof. Let $L_2 = \Sigma^*$ we have that $\forall L, L \subseteq \Sigma^*$, thus $L_1 \subseteq L_2$.

Key takeaways

- Any language (e.g., not-regular, or not-context free) is "between" two regular languages

\[
\emptyset \subset L \subset \Sigma^*
\]

- There exists a regular language such that no other language is strictly smaller
- Conversely, there exists a language such that no other language is strictly larger
Exercise 5

A Turing machine's head can stay in the same position in two consecutive steps.
Exercise 5

A Turing machine's head can stay in the same position in two consecutive steps.

Solution

True. If the head of the tape is on the beginning of the tape (left-most) and moves left, the head remains in the same position in the following step.
Exercise 6

- \(A, B\) regular
- \(C, D\) not-regular + context-free
- \(E, F\) not-regular + not-context-free

Check all that apply, or use a question-mark if it is not possible to know, in which case give a positive and a negative example.

<table>
<thead>
<tr>
<th></th>
<th>(A)</th>
<th>(C)</th>
<th>(E)</th>
<th>(A \cup B)</th>
<th>(A \cup C)</th>
<th>(C \cup D)</th>
<th>(D \cup E)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Regular</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Not-regular</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Context-free</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Not-context free</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Turing-Recognizable</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Exercise 7

- \(A, B\) regular
- \(C, D\) not-regular + context-free
- \(E, F\) not-regular + not-context-free

<table>
<thead>
<tr>
<th></th>
<th>(A)</th>
<th>(C)</th>
<th>(E)</th>
<th>(A \cup B)</th>
<th>(A \cup C)</th>
<th>(C \cup D)</th>
<th>(D \cup E)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Regular</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td></td>
<td>?</td>
<td>?</td>
<td>?</td>
</tr>
<tr>
<td>Not-regular</td>
<td></td>
<td>✓</td>
<td></td>
<td></td>
<td>?</td>
<td>?</td>
<td>?</td>
</tr>
<tr>
<td>Context-free</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>?</td>
</tr>
<tr>
<td>Not-context free</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>?</td>
</tr>
</tbody>
</table>

- \(A \cup B: \emptyset \cup \{0^n1^n \mid n \geq 0\}\) not regular; \(\Sigma^* \cup \{0^n1^n \mid n \geq 0\}\) regular
- \(C \cup D: \{0^i1^j \mid i < j\} \cup \{0^i1^j \mid i \geq j\} = \mathcal{L}(0^*1^*)\) regular;
  \(\{0^i1^j \mid i < j\} \cup \{0^n1^n \mid n \geq 0\} = \{0^i1^j \mid i \leq j\}\)
- Same idea for \(D \cup E\)
Tip

∪ on REG/CTX is akin to + on INT/LONG

- INT + LONG = LONG, similarly, REG ∪ CTX = CTX
- INT + INT = INT, similarly REG ∪ REG = REG
- Any INT is a LONG, similarly any REG is CTX

The analogy breaks with not-reg/not-ctx
Exercise 8

Spot the error

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>Regular</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Not-regular</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Context-free</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Not-context free</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Turing-Recognizable</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
</tbody>
</table>
Exercise 8

Spot the error

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>Regular</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Not-regular</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Context-free</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Not-context free</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Turing-Recognizable</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
</tbody>
</table>

Solution

<table>
<thead>
<tr>
<th>Language</th>
<th>Justification</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>A language cannot be regular and not regular</td>
</tr>
<tr>
<td>B</td>
<td>Regular/CF language is Turing recognizable</td>
</tr>
<tr>
<td>C</td>
<td>A regular language has to be CF</td>
</tr>
<tr>
<td>D</td>
<td>A language cannot be CF and not-CF</td>
</tr>
</tbody>
</table>

- We have not learned about non-Turing recognizable languages (Module 3), so they won't show up in Module 2
Exercise 9

Prove or disprove the following statement.

If $L$ is not regular, then $L$ is not context-free.
Exercise 9

Prove or disprove the following statement.

If $L$ is not regular, then $L$ is not context-free.

**Answer:** False. $\{0^n 1^n \mid n \geq 0\}$ is not-regular, yet context-free (lesson 7).
What does disproving a statement work?

Disproving $P$ means showing that $\neg P$ holds, i.e., having $P$ leads to contradiction.

Variable $\_0n1n : \text{lang.}$
Axiom $\_0n1n\_not\_reg: \neg \text{Reg } \_0n1n.$
Axiom $\_0n1n\_cf: \text{CtxFree } \_0n1n.$
Lemma not_reg_to_not_cf:
$\neg (\forall l, \neg \text{Reg } l \implies \neg \text{CtxFree } l).$

Proof.
(* Proof follows by contradiction *)
intros N. (* Assume $N: \forall l, \neg \text{Reg } l \implies \neg \text{CtxFree } l$ *)
assert (N := N \_0n1n). (* Let us instantiate $l$ to be $0^n 1^n$ in $N$ (contra-example) *)
assert (H := \_0n1n\_not\_reg). (* We know that $H: 0^n 1^n$ not regular *)
apply N in H. (* Hence, $0^n 1^n$ not context-free *)
contradict H. (* Which we contradict... *)
apply \_0n1n\_cf. (* ... because $0^n 1^n$ is context-free. *)
Qed.
What does disproving a statement work?

Essentially, disproving $P$ actually means showing that $\neg \forall x, P(x)$ holds and the goal is giving a counter-example $x$ such that $P(x)$ leads to a contradiction.

- $\neg \forall x, P(x) \iff \exists x, \neg P(x)$
- $\exists x, \neg P(x)$ and since we know $P(x)$, then we reach a contradiction
Exercise 10: rejects abb

```
a, \epsilon \rightarrow a
b, \epsilon \rightarrow b
\epsilon, \epsilon \rightarrow \epsilon
a, \epsilon \rightarrow \epsilon
b, \epsilon \rightarrow \epsilon
```

```
a, a \rightarrow \epsilon
b, b \rightarrow \epsilon
```

```
\epsilon, \$, \rightarrow \epsilon
```

```
start \rightarrow q_1 \rightarrow q_2 \rightarrow q_3
```
Exercise 10: rejects abb
Example 11

- Give the configuration history for the smallest string the TM accepts
- Given configuration $XC0Y1$, what is the next configuration
Example 11

- Give the configuration history for the smallest string the TM accepts:
  $S \square$
  $q_{\text{accept}} \square$
- Given configuration $XC0Y1$, what is the next configuration: $CX0Y1$