Extended Introduction to Computer Science
CS1001.py

Module H Text Related Algorithms: Context Free Grammars and CYK Parsing Algorithm

Instructors: Elhanan Borenstein, Michal Kleinbort

Teaching Assistants: Noam Parzanchevsky, Asaf Cassel, Shaked Dovrat, Omri Porat

School of Computer Science
Tel-Aviv University
Spring Semester 2021
http://tau-cs1001-py.wikidot.com

* Some Slides are based on an old version of the "computational models" course

And Now For Something Completely Different*,**

image from: https://www.bbc.com/news/health-54027269

* שקף עם כותרת זו יישמש להדגשת המעבר בין חלקים שונים בקורס. מי שהלכו קצת לאיבוד, זו הזדמנות לקפוץ חזרה על הרכבת.

** אנו מזמינים אתכם לשלוח לנו הצעות לתמונות שיופיעו על שקפים אלו

Module G - Text Related Algorithms: Overview

- CYK parsing algorithm (this lecture)
- Text compression
  - Huffman compression
  - Lempel-Ziv compression

Lecture Plan

- Grammars (reminder)
- Context-free grammars (CFG)
- CFG's in Chomsky normal form (CNF)
- Parsing a string (CYK algorithm)

Grammars (reminder)

- The syntax of a programming language is formally defined using a Grammar
  - Similar to the case of Natural Language, yet with much less irregularities
- Before evaluating your program, the interpreter (e.g. IDLE) verifies that it conforms to the Grammar

Example (reminder)

<SENTENCE> → <NP> <VERB>

<NP> → <ARTICLE> <NOUN>

<NOUN> → boy | girl | cat

<ARTICLE> → a | the

<VERB> → touches | likes | sees

<ARTICLE> → a
<ARTICLE> → the

Variable or Non-terminal

Production rule

Terminal

Which of the following strings can be formed using this grammar?

"a girl likes" ✓

"the boy sees" ✓

"the girl likes the cat" ✗

Context-Free Grammars (CFG): Motivation

- In this lecture we focus on an important, restricted type of grammar called Context-Free Grammar (CFG).

- Context-free grammars arise in linguistics and natural language processing (NLP), where they are used to describe the structure of sentences and words in a natural language.

Context Free Grammars (CFG)

- In a context free grammar, each rule has the form:

  <single_variable> → …

  - That is, LHS is always a single variable

- In words, rules can be applied regardless of the context of a variable. No matter which symbols surround it, the single variable on the left-hand side can always be replaced by the right-hand side.

- This is what distinguishes it from context-sensitive grammars, which allow rules of the form:

  $$xAy \rightarrow xBy$$

  - A can be replaced by B only if it appears "in the context" xAy

CFG: Example 1

Grammar G_1:

Variables = $\{A, B\}$

Terminals = $\{0,1,\#\}$

Rules:
- $A \rightarrow 0A1$
- $A \rightarrow B$
- $B \rightarrow \#$

Start Variable: A

Derivation with G_1:

$$\begin{aligned} \mathbf{A} &\rightarrow 0A1 \\ &\rightarrow 00A11 \\ &\rightarrow 000A111 \\ &\rightarrow 000B111 \\ &\rightarrow \mathbf{000\#111} \end{aligned}$$

Question 1

What strings can be generated in this way from the grammar G_1?

Answer: Exactly those of the form $0^n\#1^n$ $(n \geq 0)$.

Generating Strings from CFG

1. Write down the start variable.

2. Repeat until no variables remain.
    1. Pick a variable X written down in current string and a derivation $X \rightarrow w$.
    2. Replace X with w.

3. Return final string (must contain only terminals)

CFG: Formal Definitions

A context-free grammar is a 4-tuple (V, Σ, R, S), where

- V is a finite set of variables
- Σ is a finite set of terminals $(V \cap \Sigma = \emptyset)$
- R is a finite set of rules of the form $A \to x$, where $A \in V$ and $x \in (V \cup \Sigma)^*$.
- $S \in V$ is the start symbol.

- Let $u, v \in (V \cup \Sigma)^*$. If $(A \to w) \in R$, then uAv yields uwv, denoted $uAv \to uwv$.
- $u \xrightarrow{*} v$ if $u = v$, or $u \to u_1 \to \ldots \to u_k \to v$ for some sequence $u_1, u_2, \ldots, u_k$

Note that if $A \xrightarrow{*} xBy$ and $B \xrightarrow{*} z$, then $A \xrightarrow{*} xzy$.

The language of the grammar G, denoted $\mathcal{L}(G)$, is $\{w \in \Sigma^* : S \xrightarrow{*} w\}$

where $\xrightarrow{*}$ is determined by G.

CFG: Example 2

Grammar G_2:

Variables = $\{S\}$

Terminals = $\{a, b\}$

Rules: $S \rightarrow aSb \mid SS \mid \epsilon$

Start Variable: S

$(\epsilon = empty\ string)$

Some words in the language: *aabb*, *aababb*.

Question 3

What is this language?

Hint: Think of parentheses: i.e., *a* is "(" and *b* is ")". (()), (()())

Using larger alphabet (i.e., more terminals), ([]()), represent well formed programs with many kinds of nested loops, "if then/else" statements.

Parse Trees: Example 1

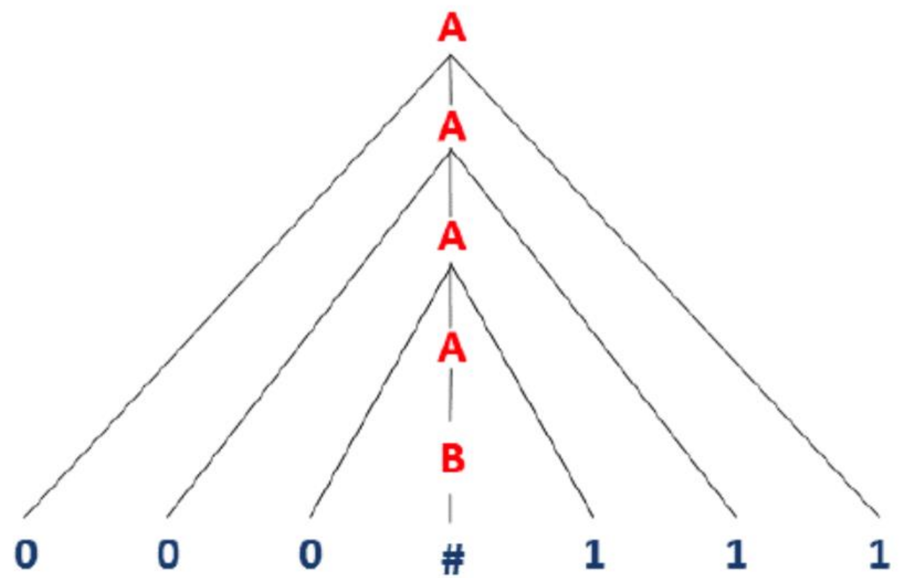
Grammar G_1:

Variables = $\{A, B\}$

Terminals = $\{0, 1, \#\}$

Rules:
- $A \rightarrow 0A1$
- $A \rightarrow B$
- $B \rightarrow \#$

Start Variable: A



The (unique) derivation sequence from A to $000\#111$:
$A \rightarrow 0A1 \rightarrow 00A11 \rightarrow 000A111 \rightarrow 000B111 \rightarrow 000\#111$

Parse Trees

Definition 10 (parse tree)

A labeled tree T is a parse tree of CFG $G = (V, \Sigma, R, S)$, if

- Inner nodes labels by elements of V.
- Leaves are labeled by elements of $V \cup \Sigma \cup \{\varepsilon\}$.
- If $n_1, \ldots, n_k$ are the labels of the direct descendants (from left to right) of a node labeled A, then $(A \to n_1, \ldots, n_k) \in R$

The yield of T, are the labels of all its leaves ordered written from left to right.

Parse Trees: Example 2

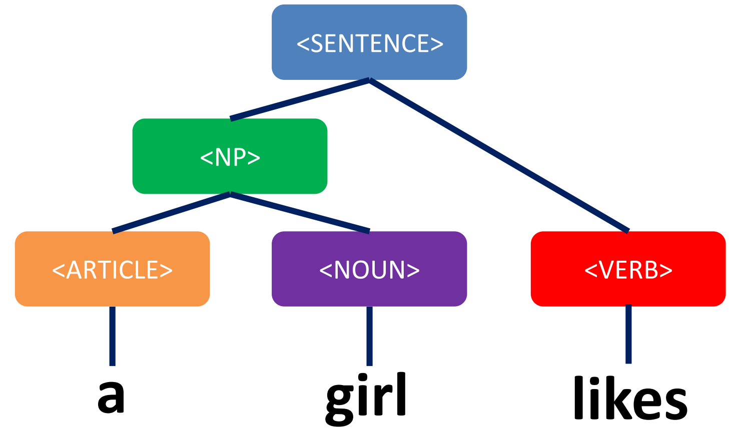
<SENTENCE> → <NP> <VERB>

<NP> → <ARTICLE> <NOUN>

<NOUN> → boy | girl | cat

<ARTICLE> → a | the

<VERB> → touches | likes | sees



- The same tree can represent various derivation orders:

<SENTENCE> → <NP> <VERB> → <ARTICLE> <NOUN> <VERB>
→ a <NOUN> <VERB>
→ a girl <VERB>
→ a girl likes

<SENTENCE> → <NP> <VERB> → <NP> likes
→ <ARTICLE> <NOUN> likes
→ a <NOUN> likes
→ a girl likes

Parse Trees: Example 2

- It may be the case for a string to have more than a single derivation sequence, represented by the same tree.
- For example:

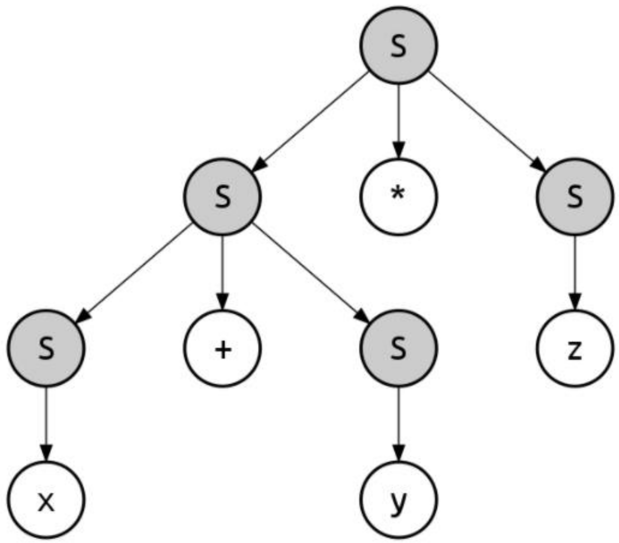
Grammar G_3:

Variables = $\{S\}$

Terminals = $\{x, y, z, +, *\}$

Rules: $S \rightarrow S * S \mid S + S \mid x \mid y \mid z$

Start Variable: S



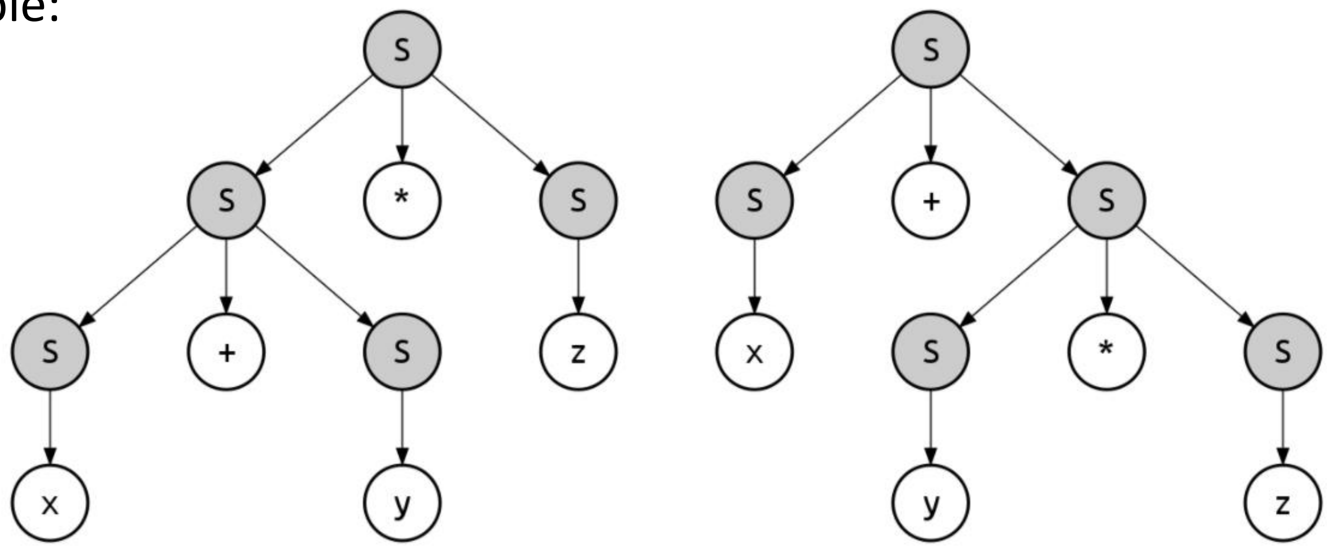
- 2 derivation sequences for $x + y * z$:
  - $S \rightarrow S * S \rightarrow S + S * S \rightarrow x + S * S \rightarrow x + y * S \rightarrow x + y * z$
  - $S \rightarrow S * S \rightarrow S + S * S \rightarrow S + S * z \rightarrow x + S * z \rightarrow x + y * z$

- Here, the only difference is the order in which we apply the rules

Parse Trees: Example 2

- But it may be the case for a string to have more than a single parse tree.

- For example:



- $S \rightarrow S * S \rightarrow S + S * S \rightarrow x + S * S \rightarrow x + y * S \rightarrow x + y * z$
- $S \rightarrow S * S \rightarrow S + S * S \rightarrow S + S * z \rightarrow x + S * z \rightarrow x + y * z$
- $S \rightarrow S + S \rightarrow S + S * S \rightarrow S + S * z \rightarrow x + S * z \rightarrow x + y * z$

- This ambiguity raises a problem: the same string may have more than a single evaluation. We will not deal with this here.

Checking Membership

Challenge

Given a CFG G and a string w, decide whether $w \in \mathcal{L}(G)$?

Initial Idea: Design an algorithm that tries all derivations.

Problem: If G does not generate w, we'll never stop.

Possible solution: Use special grammars that are:

- just as expressive!
- better for checking membership.

Chomsky Normal Form (CNF)

CFG's in Chomsky Normal Form (CNF)

A simplified, canonical form of context free grammars.

$G = (V, \Sigma, R, S)$ is in a CNF, if every rule in R has one of the following forms:

$$\begin{array}{lll} A \to a, & A \in V \;\wedge\; a \in \Sigma \\ A \to BC, & A \in V \;\wedge\; B, C \in V \setminus \{S\} \\ S \to \varepsilon. & \end{array}$$

Simpler to analyze: each derivation adds (at most) a single terminal, S only appears once, ε appears only at the empty word

What does parse tree look like?

"Most" internal nodes are of degree 2, except parents of leaves that have degree 1.

Generality of CNF

- **Theorem** (proof omitted):

  Any CFG can be converted into an equivalent grammar in Chomsky Normal Form.

- Disadvantages of CNF: (1) may require more variables & rules (2) may be artificial: relation to problem domain less obvious

- Advantages of CNF : simpler, allows more rigorous mathematical study and the construction of efficient parsing algorithms (one of which we will see now)

CYK Parsing Algorithm
For CFG in CNF

Recognition & Parsing

- A context-free grammar G defines a context-free language $L(G)$
- We would like to be able to test whether a string st belongs to the language $L(G)$ (aka recognition, or membership checking problem)
- Sometimes we are also interested in producing a parse tree for st (parsing)
  - This is more costly
  - st may have many(!) parses (this is termed ambiguity)

The CYK Algorithm

- Named after Cocke-Younger-Kasami
- A recognition and parsing algorithm suitable for languages defined by context-free grammars in CNF
- Uses “dynamic programming”: Composes the solutions from solutions to sub-problems that are stored in a table
- Instead of recursion, it works in a bottom-up fashion

CYK: Description

- Let $n = |st|$
- The algorithm constructs an $n \times (n+1)$ table T, whose cells represent all possible consecutive substrings of st
  - cell $T[i][j]$ represents the substring $st[i:j]$
  - Thus, cell $T[0][n]$ represents the whole string $st[0:n] = st$
- We assign a variable A to cell $T[i][j]$ if $st[i:j]$ can be generated from A
- The algorithm will first consider all substrings of length 1 (where $j = i + 1$), then substrings of length 2, and so on

CYK: Description (steps)

- Recall that the non-terminal production rules are of the form $A \to BC$ (rules are in CNF form).

- For cells representing a substring $st[i:j]$ of length ≥ 2 we assign a variable A if both:
  - $A \to BC$ is a production rule
  - $st[i:j]$ can be partitioned into two non-empty substrings $st[i:k], st[k:j]$, such that:
    - $st[i:k]$ can be generated from variable B and
    - $st[k:j]$ can be generated from variable C

CYK: Description (termination)

- The algorithm returns True if and only if the cell representing the entire string $st[0{:}\,n]$ contains the start symbol S
  - This means that st can be generated from variable S

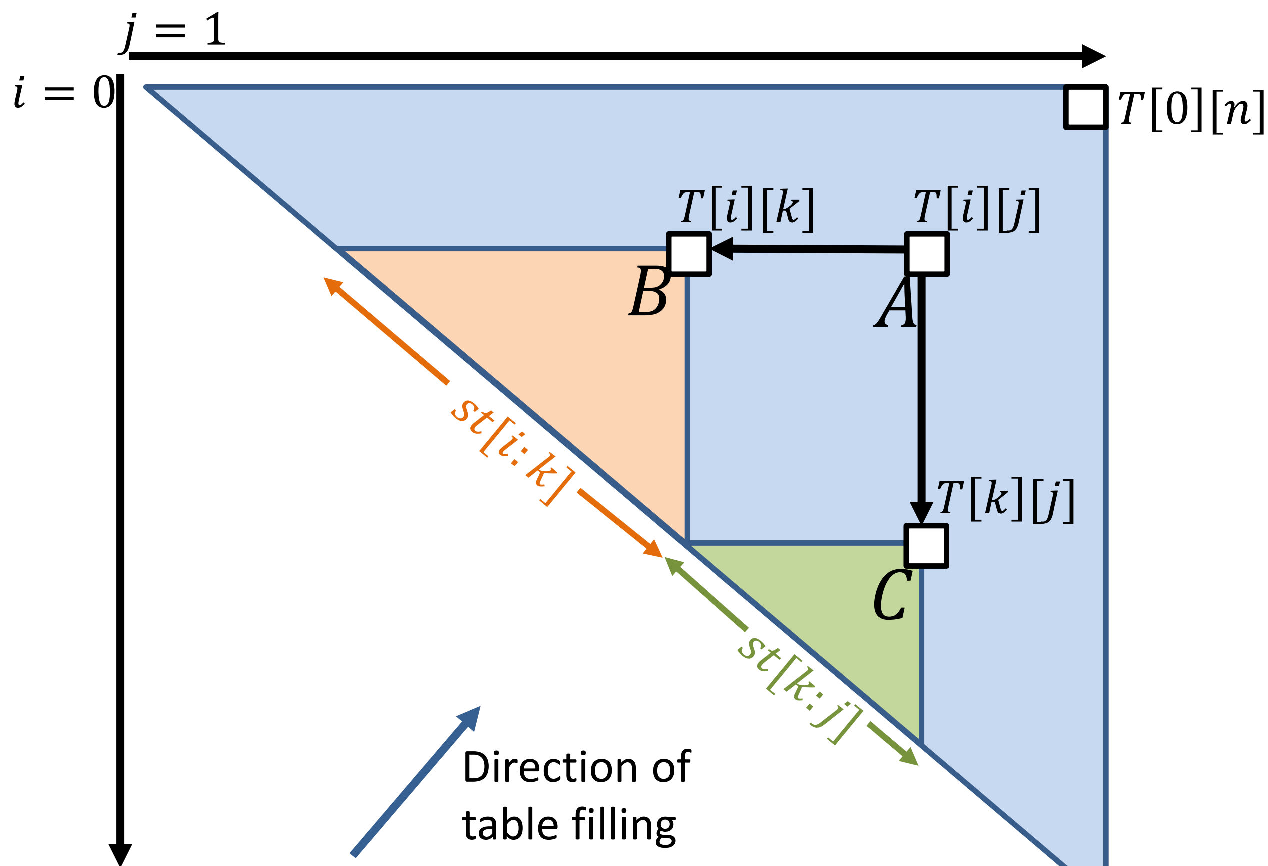
Assigning Variables to Cell $T[i][j]$

- $T[i][j]$ represents $st[i:j]$
- There are $j - i - 1$ possible non-trivial partitions of $st[i:j]$ into $st[i:k], st[k:j]$
- According to the order of computation the corresponding cells of $T[i][k], T[k][j]$ were already assigned with variables
- Therefore, we can construct the solution to $T[i][j]$ using solutions to sub-problems $T[i][k], T[k][j]$, for all $i < k < j$

The Table T

$T[i][j]$ contains any variables which can derive $st[i:j]$

$$A \rightarrow BC$$



Example

Consider the following grammar:

Variables = $\{S, A, B, C\}$

Terminals = $\{a, b\}$

Rules:

- $S \rightarrow AB \mid BC$
- $A \rightarrow BA \mid a$
- $B \rightarrow CC \mid b$
- $C \rightarrow AB \mid \mathrm{a}$

Start Variable: S

Does $st = \mathrm{baaba}$ belongs to the language $L(G)$?

$S \to AB \mid BC$
$A \to BA \mid a$
$B \to CC \mid b$
$C \to AB \mid a$

Example: “baaba”

$i \setminus$ j	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$						
$i = 1$						
$i = 2$						
$i = 3$						
$i = 4$						

$S \to AB \mid BC$
$A \to BA \mid a$
$B \to CC \mid b$
$C \to AB \mid a$

Example: “baaba”

$i \setminus j$	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$						
$i = 1$						
$i = 2$						
$i = 3$						
$i = 4$						

$S \rightarrow AB \mid BC$
$A \rightarrow BA \mid a$
$B \rightarrow CC \mid b$
$C \rightarrow AB \mid a$

Example: “baaba”

$i \setminus$ j	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		“b”				
$i = 1$			“a”			
$i = 2$				“a”		
$i = 3$					“b”	
$i = 4$						“a”

$S \to AB \mid BC$
$A \to BA \mid a$
$B \to CC \mid b$
$C \to AB \mid a$

Example: “baaba”

$i \setminus$ j	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		“b” {B}				
$i = 1$			“a”			
$i = 2$				“a”		
$i = 3$					“b”	
$i = 4$						“a”

$S \to AB \mid BC$
$A \to BA \mid a$
$B \to CC \mid b$
$C \to AB \mid a$

Example: “baaba”

$i \setminus j$	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		“b” {B}				
$i = 1$			“a” {A,C}			
$i = 2$				“a”		
$i = 3$					“b”	
$i = 4$						“a”

$S \rightarrow AB \mid BC$
$A \rightarrow BA \mid a$
$B \rightarrow CC \mid b$
$C \rightarrow AB \mid a$

Example: “baaba”

$i \backslash$ j	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		“b” {B}				
$i = 1$			“a” {A,C}			
$i = 2$				“a” {A,C}		
$i = 3$					“b”	
$i = 4$						“a”

$S \rightarrow AB \mid BC$
$A \rightarrow BA \mid a$
$B \rightarrow CC \mid b$
$C \rightarrow AB \mid a$

Example: “baaba”

$i \setminus j$	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		“b” {B}				
$i = 1$			“a” {A,C}			
$i = 2$				“a” {A,C}		
$i = 3$					“b” {B}	
$i = 4$						“a”

$S \to AB \mid BC$
$A \to BA \mid a$
$B \to CC \mid b$
$C \to AB \mid a$

Example: “baaba”

$i \setminus$ j	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		“b” {B}				
$i = 1$			“a” {A,C}			
$i = 2$				“a” {A,C}		
$i = 3$					“b” {B}	
$i = 4$						“a” {A,C}

$S \to AB \mid BC$
$A \to BA \mid a$
$B \to CC \mid b$
$C \to AB \mid a$

Example: “baaba”

$i \setminus j$	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		“b” {B}	“ba”			
$i = 1$			“a” {A,C}	“aa”		
$i = 2$				“a” {A,C}	“ab”	
$i = 3$					“b” {B}	“ba”
$i = 4$						“a” {A,C}

$S \rightarrow AB \mid BC$
$A \rightarrow BA \mid a$
$B \rightarrow CC \mid b$
$C \rightarrow AB \mid a$

Example: “baaba”

$i \setminus$ j	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		“b” {B}	“ba”			
$i = 1$			“a” {A,C}	“aa”		
$i = 2$				“a” {A,C}	“ab”	
$i = 3$					“b” {B}	“ba”
$i = 4$						“a” {A,C}

$S \rightarrow AB \mid BC$
$A \rightarrow BA \mid a$
$B \rightarrow CC \mid b$
$C \rightarrow AB \mid a$

Example: “baaba”

$i \setminus j$	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		“b” {B}	“ba”			
$i = 1$			“a” {A,C}			
$i = 2$						
$i = 3$						
$i = 4$						

Computing $T[0][2]$:

$$T[0][1] = \{B\}$$
$$T[1][2] = \{A, C\}$$

Is there a rule with r.h.s BA?
- Yes! $A \rightarrow BA$
  then add A

Is there a rule with r.h.s BC?
- Yes! $S \rightarrow BC$
  then add S

$S \to AB \mid BC$
$A \to BA \mid a$
$B \to CC \mid b$
$C \to AB \mid a$

Example: "baaba"

$i \setminus$ j	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		"b" {B}	"ba" {A,S}			
$i = 1$			"a" {A,C}	"aa"		
$i = 2$				"a" {A,C}	"ab"	
$i = 3$					"b" {B}	"ba"
$i = 4$						"a" {A,C}

$S \to AB \mid BC$
$A \to BA \mid a$
$B \to CC \mid b$
$C \to AB \mid a$

Example: “baaba”

$i \setminus j$	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		“b” {B}	“ba” {A,S}			
$i = 1$			“a” {A,C}	“aa”		
$i = 2$				“a” {A,C}	“ab”	
$i = 3$					“b” {B}	“ba”
$i = 4$						“a” {A,C}

$S \to AB \mid BC$
$A \to BA \mid a$
$B \to CC \mid b$
$C \to AB \mid a$

Example: “baaba”

$i \setminus$ j	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		“b” {B}	“ba” {A,S}			
$i = 1$			“a” {A,C}	“aa”		
$i = 2$				“a” {A,C}	“ab”	
$i = 3$					“b” {B}	“ba”
$i = 4$						“a” {A,C}

$S \rightarrow AB \mid BC$
$A \rightarrow BA \mid a$
$B \rightarrow CC \mid b$
$C \rightarrow AB \mid a$

Example: “baaba”

i \ j	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		“b” {B}	“ba” {A,S}			
$i = 1$			“a” {A,C}	“aa”		
				“a” {A,C}	“ab”	
					“b” {B}	“ba”
						“a” {A,C}

Computing $T[1][3]$:

$$T[1][2] = \{A, C\}$$
$$T[2][3] = \{A, C\}$$

Is there a rule with r.h.s AA?
- No.

No rules with r.h.s AC or CA either
Is there a rule with r.h.s CC?
- Yes! $B \rightarrow CC$
  then add B

$S \to AB \mid BC$
$A \to BA \mid a$
$B \to CC \mid b$
$C \to AB \mid a$

Example: “baaba”

$i \setminus$ j	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		“b” {B}	“ba” {A,S}			
$i = 1$			“a” {A,C}	“aa” {B}		
$i = 2$				“a” {A,C}	“ab”	
$i = 3$					“b” {B}	“ba”
$i = 4$						“a” {A,C}

$S \to AB \mid BC$
$A \to BA \mid a$
$B \to CC \mid b$
$C \to AB \mid a$

Example: “baaba”

$i \setminus j$	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		“b” {B}	“ba” {A,S}			
$i = 1$			“a” {A,C}	“aa” {B}		
$i = 2$				“a” {A,C}	“ab”	
$i = 3$					“b” {B}	“ba”
$i = 4$						“a” {A,C}

$S \to AB \mid BC$
$A \to BA \mid a$
$B \to CC \mid b$
$C \to AB \mid a$

Example: "baaba"

$i \setminus j$	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		"b" {B}	"ba" {A,S}			
$i = 1$			"a" {A,C}	"aa" {B}		
$i = 2$				"a" {A,C}	"ab" {S,C}	
$i = 3$					"b" {B}	"ba"
$i = 4$						"a" {A,C}

$S \to AB \mid BC$
$A \to BA \mid a$
$B \to CC \mid b$
$C \to AB \mid a$

Example: "baaba"

$i \backslash$ j	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		"b" {B}	"ba" {A,S}			
$i = 1$			"a" {A,C}	"aa" {B}		
$i = 2$				"a" {A,C}	"ab" {S,C}	
$i = 3$					"b" {B}	"ba"
$i = 4$						"a" {A,C}

$S \rightarrow AB \mid BC$
$A \rightarrow BA \mid a$
$B \rightarrow CC \mid b$
$C \rightarrow AB \mid a$

Example: “baaba”

$i \setminus j$	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		“b” {B}	“ba” {A,S}			
$i = 1$			“a” {A,C}	“aa” {B}		
$i = 2$				“a” {A,C}	“ab” {S,C}	
$i = 3$					“b” {B}	“ba” {A,S}
$i = 4$						“a” {A,C}

$S \to AB \mid BC$
$A \to BA \mid a$
$B \to CC \mid b$
$C \to AB \mid a$

Example: "baaba"

$i \backslash$ j	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		"b" {B}	"ba" {A,S}	"baa"		
$i = 1$			"a" {A,C}	"aa" {B}	"aab"	
$i = 2$				"a" {A,C}	"ab" {S,C}	"aba"
$i = 3$					"b" {B}	"ba" {A,S}
$i = 4$						"a" {A,C}

$S \to AB \mid BC$
$A \to BA \mid a$
$B \to CC \mid b$
$C \to AB \mid a$

Example: “baaba”

$i \setminus$ j	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		“b” {B}	“ba” {A,S}	“baa”		
$i = 1$			“a” {A,C}	“aa” {B}	“aab”	
$i = 2$				“a” {A,C}	“ab” {S,C}	“aba”
$i = 3$					“b” {B}	“ba” {A,S}
$i = 4$						“a” {A,C}

$S \rightarrow AB \mid BC$
$A \rightarrow BA \mid a$
$B \rightarrow CC \mid b$
$C \rightarrow AB \mid a$

Example: “baaba”

$i \setminus j$	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		“b” {B}	“ba” {A,S}	“baa”		
$i = 1$			“a” {A,C}	“aa” {B}	“aab”	
				“a” {A,C}	“ab” {S,C}	“aba”
					“b” {B}	“ba” {A,S}
$i = 4$						“a” {A,C}

Computing $T[0][3]$:
- Option #1: use

$$T[0][1] = \{B\}$$
$$T[1][3] = \{B\}$$

No rule with r.h.s BB.

$S \to AB \mid BC$
$A \to BA \mid a$
$B \to CC \mid b$
$C \to AB \mid a$

Example: “baaba”

$i \setminus$ j	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		“b” {B}	“ba” {A,S}	“baa”		
$i = 1$			“a” {A,C}	“aa” {B}	“aab”	
				“a” {A,C}	“ab” {S,C}	“aba”
					“b” {B}	“ba” {A,S}
						“a” {A,C}

Computing $T[0][3]$:
- Option #1: use

$$T[0][1] = \{B\}$$
$$T[1][3] = \{B\}$$

No rule with r.h.s BB.
- Option #2: use

$$T[0][2] = \{A, S\}$$
$$T[2][3] = \{A, C\}$$

No rules with r.h.s AA, AC, SA, SC

$S \rightarrow AB \mid BC$
$A \rightarrow BA \mid a$
$B \rightarrow CC \mid b$
$C \rightarrow AB \mid a$

Example: “baaba”

$i \setminus$ j	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		“b” {B}	“ba” {A,S}	“baa” ∅		
$i = 1$			“a” {A,C}	“aa” {B}	“aab”	
$i = 2$				“a” {A,C}	“ab” {S,C}	“aba”
$i = 3$					“b” {B}	“ba” {A,S}
$i = 4$						“a” {A,C}

$S \rightarrow AB \mid BC$
$A \rightarrow BA \mid a$
$B \rightarrow CC \mid b$
$C \rightarrow AB \mid a$

Example: “baaba”

$i \setminus$ j	$j=0$	$j=1$	$j=2$	$j=3$	$j=4$	$j=5$
$i=0$		“b” {B}	“ba” {A,S}	“baa” ∅		
$i=1$			“a” {A,C}	“aa” {B}	“aab”	
				“a” {A,C}	“ab” {S,C}	“aba”
$i=3$					“b” {B}	“ba” {A,S}
$i=4$						“a” {A,C}

Computing $T[1][4]$:
- Option #1: Add B (since $B \rightarrow CC$)
  No rules with r.h.s AS,AC or CS.

$S \to AB \mid BC$
$A \to BA \mid a$
$B \to CC \mid b$
$C \to AB \mid a$

Example: “baaba”

$i \setminus j$	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		“b” {B}	“ba” {A,S}	“baa” ∅		
$i = 1$			“a” {A,C}	“aa” {B}	“aab”	
				“a” {A,C}	“ab” {S,C}	“aba”
					“b” {B}	“ba” {A,S}
$i = 4$						“a” {A,C}

Computing $T[1][4]$:
- Option #1: Add B (since $B \to CC$)
  No rules with r.h.s AS,AC or CS.
- Option #2: No rule with r.h.s BB.

$S \to AB \mid BC$
$A \to BA \mid a$
$B \to CC \mid b$
$C \to AB \mid a$

Example: “baaba”

$i \setminus j$	$j=0$	$j=1$	$j=2$	$j=3$	$j=4$	$j=5$
$i=0$		“b” {B}	“ba” {A,S}	“baa” Ø		
$i=1$			“a” {A,C}	“aa” {B}	“aab” {B}	
$i=2$				“a” {A,C}	“ab” {S,C}	“aba”
$i=3$					“b” {B}	“ba” {A,S}
$i=4$						“a” {A,C}

$S \rightarrow AB \mid BC$
$A \rightarrow BA \mid a$
$B \rightarrow CC \mid b$
$C \rightarrow AB \mid a$

Example: “baaba”

$i \setminus$ j	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		“b” {B}	“ba” {A,S}	“baa” Ø		
$i = 1$			“a” {A,C}	“aa” {B}	“aab” {B}	
$i = 2$				“a” {A,C}	“ab” {S,C}	“aba”
$i = 3$					“b” {B}	“ba” {A,S}
$i = 4$						“a” {A,C}

$S \to AB \mid BC$
$A \to BA \mid a$
$B \to CC \mid b$
$C \to AB \mid a$

Example: “baaba”

$i \setminus$ j	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		“b” {B}	“ba” {A,S}	“baa” ∅		
$i = 1$			“a” {A,C}	“aa” {B}	“aab” {B}	
$i = 2$				“a” {A,C}	“ab” {S,C}	“aba”
$i = 3$					“b” {B}	“ba” {A,S}
$i = 4$						“a” {A,C}

$S \rightarrow AB \mid BC$
$A \rightarrow BA \mid a$
$B \rightarrow CC \mid b$
$C \rightarrow AB \mid a$

Example: "baaba"

$i \setminus$ j	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		"b" {B}	"ba" {A,S}	"baa" Ø		
$i = 1$			"a" {A,C}	"aa" {B}	"aab" {B}	
$i = 2$				"a" {A,C}	"ab" {S,C}	"aba" {B}
$i = 3$					"b" {B}	"ba" {A,S}
$i = 4$						"a" {A,C}

$S \rightarrow AB \mid BC$
$A \rightarrow BA \mid a$
$B \rightarrow CC \mid b$
$C \rightarrow AB \mid a$

Example: “baaba”

$i \setminus$ j	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		“b” {B}	“ba” {A,S}	“baa” ∅		
$i = 1$			“a” {A,C}	“aa” {B}	“aab” {B}	
$i = 2$				“a” {A,C}	“ab” {S,C}	“aba” {B}
$i = 3$					“b” {B}	“ba” {A,S}
$i = 4$						“a” {A,C}

$S \rightarrow AB \mid BC$
$A \rightarrow BA \mid a$
$B \rightarrow CC \mid b$
$C \rightarrow AB \mid a$

Example: “baaba”

$i \setminus$ j	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		“b” {B}	“ba” {A,S}	“baa” ∅	“baab”	
$i = 1$			“a” {A,C}	“aa” {B}	“aab” {B}	“aaba”
$i = 2$				“a” {A,C}	“ab” {S,C}	“aba” {B}
$i = 3$					“b” {B}	“ba” {A,S}
$i = 4$						“a” {A,C}

$S \to AB \mid BC$
$A \to BA \mid a$
$B \to CC \mid b$
$C \to AB \mid a$

Example: “baaba”

$i \setminus j$	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		“b” {B}	“ba” {A,S}	“baa” Ø	“baab”	
$i = 1$			“a” {A,C}	“aa” {B}	“aab” {B}	“aaba”
$i = 2$				“a” {A,C}	“ab” {S,C}	“aba” {B}
$i = 3$					“b” {B}	“ba” {A,S}
$i = 4$						“a” {A,C}

$S \rightarrow AB \mid BC$
$A \rightarrow BA \mid a$
$B \rightarrow CC \mid b$
$C \rightarrow AB \mid a$

Example: “baaba”

i \ j	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		“b” {B}	“ba” {A,S}	“baa” ∅	“baab”	
$i = 1$			“a” {A,C}	“aa” {B}	“aab” {B}	“aaba”
$i = 2$				“a” {A,C}	“ab” {S,C}	“aba” {B}
$i = 3$					“b” {B}	“ba” {A,S}
$i = 4$						“a” {A,C}

$S \to AB \mid BC$
$A \to BA \mid a$
$B \to CC \mid b$
$C \to AB \mid a$

Example: "baaba"

$i \setminus j$	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		"b" {B}	"ba" {A,S}	"baa" Ø	"baab"	
$i = 1$			"a" {A,C}	"aa" {B}	"aab" {B}	"aaba"
$i = 2$				"a" {A,C}	"ab" {S,C}	"aba" {B}
$i = 3$					"b" {B}	"ba" {A,S}
$i = 4$						"a" {A,C}

$S \rightarrow AB \mid BC$
$A \rightarrow BA \mid a$
$B \rightarrow CC \mid b$
$C \rightarrow AB \mid a$

Example: “baaba”

$i \backslash j$	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		“b” {B}	“ba” {A,S}	“baa” ∅	“baab”	
$i = 1$			“a” {A,C}	“aa” {B}	“aab” {B}	“aaba”
$i = 2$				“a” {A,C}	“ab” {S,C}	“aba” {B}
$i = 3$					“b” {B}	“ba” {A,S}
$i = 4$						“a” {A,C}

$S \to AB \mid BC$
$A \to BA \mid a$
$B \to CC \mid b$
$C \to AB \mid a$

Example: "baaba"

$i \setminus j$	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		"b" {B}	"ba" {A,S}	"baa" ∅	"baab" ∅	
$i = 1$			"a" {A,C}	"aa" {B}	"aab" {B}	"aaba"
$i = 2$				"a" {A,C}	"ab" {S,C}	"aba" {B}
$i = 3$					"b" {B}	"ba" {A,S}
$i = 4$						"a" {A,C}

$S \to AB \mid BC$
$A \to BA \mid a$
$B \to CC \mid b$
$C \to AB \mid a$

Example: “baaba”

$i \setminus j$	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		“b” {B}	“ba” {A,S}	“baa” ∅	“baab” ∅	
$i = 1$			“a” {A,C}	“aa” {B}	“aab” {B}	“aaba”
$i = 2$				“a” {A,C}	“ab” {S,C}	“aba” {B}
$i = 3$					“b” {B}	“ba” {A,S}
$i = 4$						“a” {A,C}

$S \to AB \mid BC$
$A \to BA \mid a$
$B \to CC \mid b$
$C \to AB \mid a$

Example: "baaba"

$i \setminus j$	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		"b" {B}	"ba" {A,S}	"baa" Ø	"baab" Ø	
$i = 1$			"a" {A,C}	"aa" {B}	"aab" {B}	"aaba"
$i = 2$				"a" {A,C}	"ab" {S,C}	"aba" {B}
$i = 3$					"b" {B}	"ba" {A,S}
$i = 4$						"a" {A,C}

$S \to AB \mid BC$
$A \to BA \mid a$
$B \to CC \mid b$
$C \to AB \mid a$

Example: “baaba”

$i \setminus$ j	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		“b” {B}	“ba” {A,S}	“baa” Ø	“baab” Ø	
$i = 1$			“a” {A,C}	“aa” {B}	“aab” {B}	“aaba” {S,C}
$i = 2$				“a” {A,C}	“ab” {S,C}	“aba” {B}
$i = 3$					“b” {B}	“ba” {A,S}
$i = 4$						“a” {A,C}

$S \to AB \mid BC$
$A \to BA \mid a$
$B \to CC \mid b$
$C \to AB \mid a$

Example: “baaba”

$i \setminus j$	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		“b” {B}	“ba” {A,S}	“baa” Ø	“baab” Ø	
$i = 1$			“a” {A,C}	“aa” {B}	“aab” {B}	“aaba” {S,C}
$i = 2$				“a” {A,C}	“ab” {S,C}	“aba” {B}
$i = 3$					“b” {B}	“ba” {A,S}
$i = 4$						“a” {A,C}

$S \rightarrow AB \mid BC$
$A \rightarrow BA \mid a$
$B \rightarrow CC \mid b$
$C \rightarrow AB \mid a$

Example: “baaba”

$i \setminus j$	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		“b” {B}	“ba” {A,S}	“baa” Ø	“baab” Ø	
$i = 1$			“a” {A,C}	“aa” {B}	“aab” {B}	“aaba” {S,C,A}
$i = 2$				“a” {A,C}	“ab” {S,C}	“aba” {B}
$i = 3$					“b” {B}	“ba” {A,S}
$i = 4$						“a” {A,C}

$S \to AB \mid BC$
$A \to BA \mid a$
$B \to CC \mid b$
$C \to AB \mid a$

Example: “baaba”

$i \setminus j$	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		“b” {B}	“ba” {A,S}	“baa” Ø	“baab” Ø	
$i = 1$			“a” {A,C}	“aa” {B}	“aab” {B}	“aaba” {S,C,A}
$i = 2$				“a” {A,C}	“ab” {S,C}	“aba” {B}
$i = 3$					“b” {B}	“ba” {A,S}
$i = 4$						“a” {A,C}

$S \to AB \mid BC$
$A \to BA \mid a$
$B \to CC \mid b$
$C \to AB \mid a$

Example: “baaba”

$i \setminus$ j	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		“b” {B}	“ba” {A,S}	“baa” ∅	“baab” ∅	
$i = 1$			“a” {A,C}	“aa” {B}	“aab” {B}	“aaba” {S,C,A}
$i = 2$				“a” {A,C}	“ab” {S,C}	“aba” {B}
$i = 3$					“b” {B}	“ba” {A,S}
$i = 4$						“a” {A,C}

$S \to AB \mid BC$
$A \to BA \mid a$
$B \to CC \mid b$
$C \to AB \mid a$

Example: “baaba”

$i \setminus j$	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		“b” {B}	“ba” {A,S}	“baa” ∅	“baab” ∅	“baaba”
$i = 1$			“a” {A,C}	“aa” {B}	“aab” {B}	“aaba” {S,C,A}
$i = 2$				“a” {A,C}	“ab” {S,C}	“aba” {B}
$i = 3$					“b” {B}	“ba” {A,S}
$i = 4$						“a” {A,C}

$S \rightarrow AB \mid BC$
$A \rightarrow BA \mid a$
$B \rightarrow CC \mid b$
$C \rightarrow AB \mid a$

Example: "baaba"

i \ j	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		"b" {B}	"ba" {A,S}	"baa" ∅	"baab" ∅	"baaba"
$i = 1$			"a" {A,C}	"aa" {B}	"aab" {B}	"aaba" {S,C,A}
$i = 2$				"a" {A,C}	"ab" {S,C}	"aba" {B}
$i = 3$					"b" {B}	"ba" {A,S}
$i = 4$						"a" {A,C}

$S \rightarrow AB \mid BC$
$A \rightarrow BA \mid a$
$B \rightarrow CC \mid b$
$C \rightarrow AB \mid a$

Example: “baaba”

$i \backslash j$	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		“b” {B}	“ba” {A,S}	“baa” ∅	“baab” ∅	“baaba” {S,A}
$i = 1$			“a” {A,C}	“aa” {B}	“aab” {B}	“aaba” {S,C,A}
$i = 2$				“a” {A,C}	“ab” {S,C}	“aba” {B}
$i = 3$					“b” {B}	“ba” {A,S}
$i = 4$						“a” {A,C}

$S \to AB \mid BC$
$A \to BA \mid a$
$B \to CC \mid b$
$C \to AB \mid a$

Example: “baaba”

$i \setminus j$	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		“b” {B}	“ba” {A,S}	“baa” ∅	“baab” ∅	“baaba” {S,A,C}
$i = 1$			“a” {A,C}	“aa” {B}	“aab” {B}	“aaba” {S,C,A}
$i = 2$				“a” {A,C}	“ab” {S,C}	“aba” {B}
$i = 3$					“b” {B}	“ba” {A,S}
$i = 4$						“a” {A,C}

$S \rightarrow AB \mid BC$
$A \rightarrow BA \mid a$
$B \rightarrow CC \mid b$
$C \rightarrow AB \mid a$

Example: “baaba”

$i \backslash j$	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		“b” {B}	“ba” {A,S}	“baa” Ø	“baab” Ø	“baaba” {S,A,C}
$i = 1$			“a” {A,C}	“aa” {B}	“aab” {B}	“aaba” {S,C,A}
$i = 2$				“a” {A,C}	“ab” {S,C}	“aba” {B}
$i = 3$					“b” {B}	“ba” {A,S}
$i = 4$						“a” {A,C}

$S \to AB \mid BC$
$A \to BA \mid a$
$B \to CC \mid b$
$C \to AB \mid a$

Example: “baaba”

$i \setminus j$	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		“b” {B}	“ba” {A,S}	“baa” ∅	“baab” ∅	“baaba” {S,A,C}
$i = 1$			“a” {A,C}	“aa” {B}	“aab” {B}	“aaba” {S,C,A}
$i = 2$				“a” {A,C}	“ab” {S,C}	“aba” {B}
$i = 3$					“b” {B}	“ba” {A,S}
$i = 4$						“a” {A,C}

$S \to AB \mid BC$
$A \to BA \mid a$
$B \to CC \mid b$
$C \to AB \mid a$

Example: “baaba”

$i \setminus j$	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		“b” {B}	“ba” {A,S}	“baa” Ø	“baab” Ø	“baaba” {S,A,C}
$i = 1$						“aaba” {S,C,A}
$i = 2$						“aba” {B}
$i = 3$						“ba” {A,S}
$i = 4$						“a” {A,C}

$T[0][5]$ contains the start symbol S

$\Rightarrow$ “baaba”$\in L(G)$

$S \to AB \mid BC$
$A \to BA \mid a$
$B \to CC \mid b$
$C \to AB \mid a$

Example: "baaba"

$i \backslash j$	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		"b" {B}	"ba" {A,S}	"baa" ∅	"baab" ∅	"baaba" {S,A,C}
$i = 1$			"a" {A,C}	"aa" {B}	"aab" {B}	"aaba" {S,C,A}
$i = 2$				"a" {A,C}	"ab" {S,C}	"aba" {B}
$i = 3$					"b" {B}	"ba" {A,S}
$i = 4$						"a" {A,C}

$S \to AB \mid BC$
$A \to BA \mid a$
$B \to CC \mid b$
$C \to AB \mid a$

Example: "baaba"

$i \backslash j$	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		"b" {B}	"ba" {A,S}	"baa" ∅	"baab" ∅	"baaba" {S,A,C}
$i = 1$			"a" {A,C}	"aa" {B}	"aab" {B}	"aaba" {S,C,A}
$i = 2$				"a" {A,C}	"ab" {S,C}	"aba" {B}
$i = 3$					"b" {B}	"ba" {A,S}
$i = 4$						"a" {A,C}

$S \rightarrow AB \mid BC$
$A \rightarrow BA \mid a$
$B \rightarrow CC \mid b$
$C \rightarrow AB \mid a$

Example: “baaba”

$i \setminus j$	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		“b” {B}	“ba” {A,S}	“baa” ∅	“baab” ∅	“baaba” {S,A,C}
$i = 1$			“a” {A,C}	“aa” {B}	“aab” {B}	“aaba” {S,C,A}
$i = 2$				“a” {A,C}	“ab” {S,C}	“aba” {B}
$i = 3$					“b” {B}	“ba” {A,S}
$i = 4$						“a” {A,C}

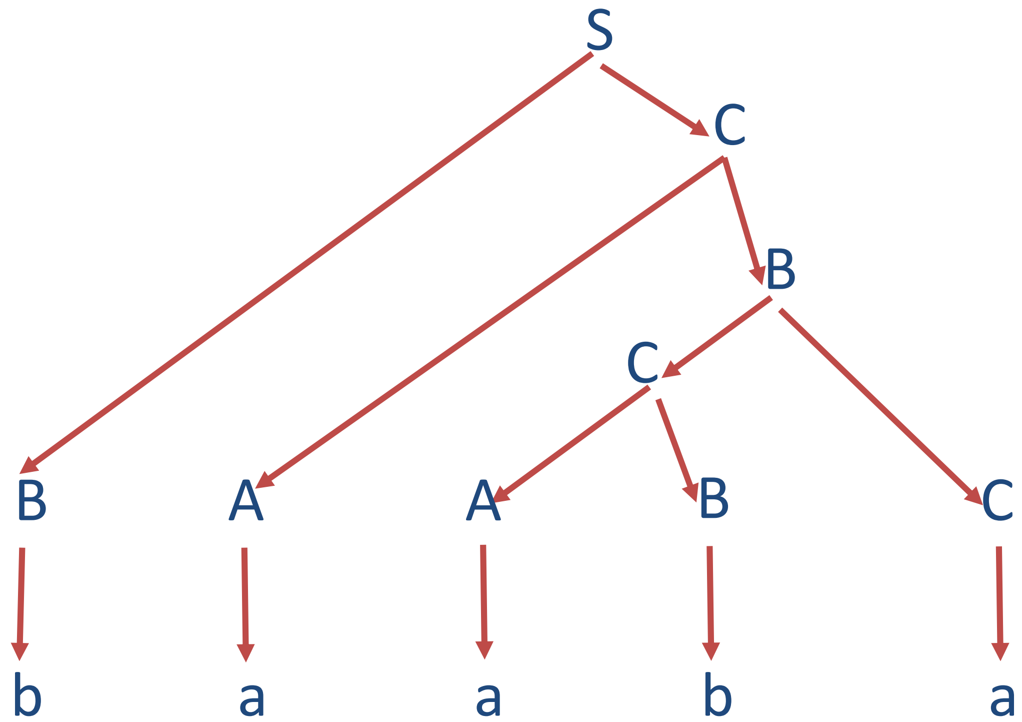
$S \rightarrow AB \mid BC$
$A \rightarrow BA \mid a$
$B \rightarrow CC \mid b$
$C \rightarrow AB \mid a$

Example: “baaba”

$i \setminus j$	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		“b” {B}	“ba” {A,S}	“baa” ∅	“baab” ∅	“baaba” {S,A,C}
$i = 1$			“a” {A,C}	“aa” {B}	“aab” {B}	“aaba” {S,C,A}
$i = 2$				“a” {A,C}	“ab” {S,C}	“aba” {B}
$i = 3$					“b” {B}	“ba” {A,S}
$i = 4$						“a” {A,C}

$S \to AB \mid BC$
$A \to BA \mid a$
$B \to CC \mid b$
$C \to AB \mid a$

Parse tree of "baaba"



$S \to AB \mid BC$
$A \to BA \mid a$
$B \to CC \mid b$
$C \to AB \mid a$

Another parse tree

$i \setminus j$	$j = 0$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
$i = 0$		"b" {B}	"ba" {A,S}	"baa" ∅	"baab" ∅	"baaba" {S,A,C}
$i = 1$			"a" {A,C}	"aa" {B}	"aab" {B}	"aaba" {S,C,A}
$i = 2$				"a" {A,C}	"ab" {S,C}	"aba" {B}
$i = 3$					"b" {B}	"ba" {A,S}
$i = 4$						"a" {A,C}

Implementation in Python

Representing the grammar in Python

Variables = $\{S, A, B, C\}$

Terminals = $\{a, b\}$

Rules:

- $S \rightarrow AB \mid BC$
- $A \rightarrow BA \mid a$
- $B \rightarrow CC \mid b$
- $C \rightarrow AB \mid$ a

Start Variable: S

Rules are represented as a ***key:value*** pair in a dict:
key = the LHS variable A of these rules
value = a set of all possible RHS of the rules of the form

$A \rightarrow$ ...

Note that RHS could either be

- of length 2, representing two variables (rule like: $A \rightarrow BA$)
- of length 1, representing a terminal (rule like: $A \rightarrow a$)

```
>>> rule_dict = {"S":{"AB","BC"},"A":{"BA","a"},"B"
:{"CC","b"}, "C":{"AB", "a"}}
>>> start_symbol = "S"
```

The variables and terminals are implicitly defined by the `rule_dict`

Running CYK Code

Variables = $\{S, A, B, C\}$
Terminals = $\{a, b\}$
Rules:

- $S \rightarrow AB \mid BC$
- $A \rightarrow BA \mid a$
- $B \rightarrow CC \mid b$
- $C \rightarrow AB \mid \mathrm{a}$

Start Variable: S

```
>>> rule_dict = {"S":{"AB","BC"},"A":{"BA","a"},"B" :{"CC","b"},
"C":{"AB", "a"}}
>>> start_symbol = "S"

>>> CYK("baaba", rule_dict, start_symbol)
True
>>> CYK("baab", rule_dict, start_symbol)
False
```

CYK Code

```
def CYK(st, rule_dict, start_var):
    n = len(st)

    # Init table for the dynamic programming algorithm
    table = [[None for j in range(n+1)] for i in range(n)]
    for i in range(n):
        for j in range(i+1,n+1):
            table[i][j] = set()

    # Fill the table cells representing substrings of length 1
    fill_length_1_cells(table, rule_dict, st)

    # Fill the table cells representing substrings of length >=2
    for length in range(2, n+1):
        for i in range(0, n-length+1):
            j = i+length
            fill_cell(table, i,j, rule_dict)

    return start_var in table[0][n]
```

CYK Code (cont.)

```python
def fill_length_1_cells(table, rule_dict, st):
    n = len(st)
    for i in range(n):
        for lhs in rule_dict: # lhs is a single variable
            if st[i] in rule_dict[lhs]
               #add variable lhs to T[i][i+1]
               table[i][i+1].add(lhs)
```

```python
def fill_cell(table, i, j, rule_dict):
    for k in range(i+1, j): # non trivial partitions of s[i:j]
        for lhs in rule_dict: # lhs is a single variable
           for rhs in rule_dict[lhs]:
               if len(rhs) == 2: # rule like A -> XY (not A -> a)
                   X, Y = rhs[0], rhs[1]
                   if X in table[i][k] and Y in table[k][j]
                       table[i][j].add(lhs)
```

Computability of Grammar-Related Problems

- Language equality: Given two CFGs, do they generate the same language?
- Language inclusion: Given two CFGs, can the first one generate all strings that the second one can?
- Grammar ambiguity: Given a CFG, is it ambiguous?
- Language disjointness: Given two CFGs, is there any string derivable from both grammars?

Can you guess what's in common to these problems?