7)

4)

Sei L regulär.

Dann gibt es ein $n \in \mathbb{N}$, so dass für alle $x \in L$ mit $|x| \geq n$ gilt:

$x = uvw$ mit:

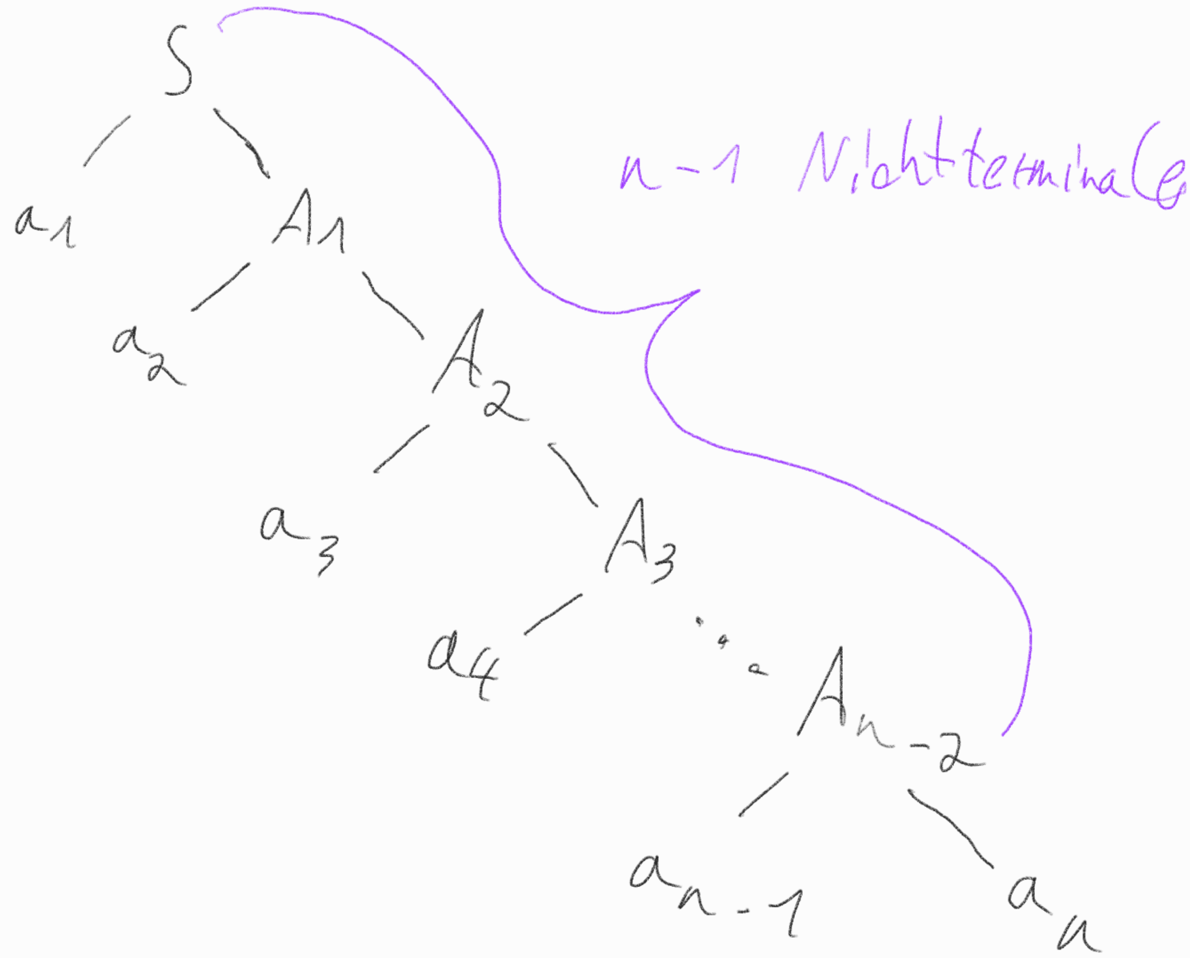
i) $|v| \geq 1$

ii) $|uv| \leq n$

iii) $uv^iw \in L$ für $i \in \mathbb{N}$

Da L regulär ist, existiert eine Typ3-Gramatik $G=(\Sigma, N, P, S)$ mit $L(G)=L$.

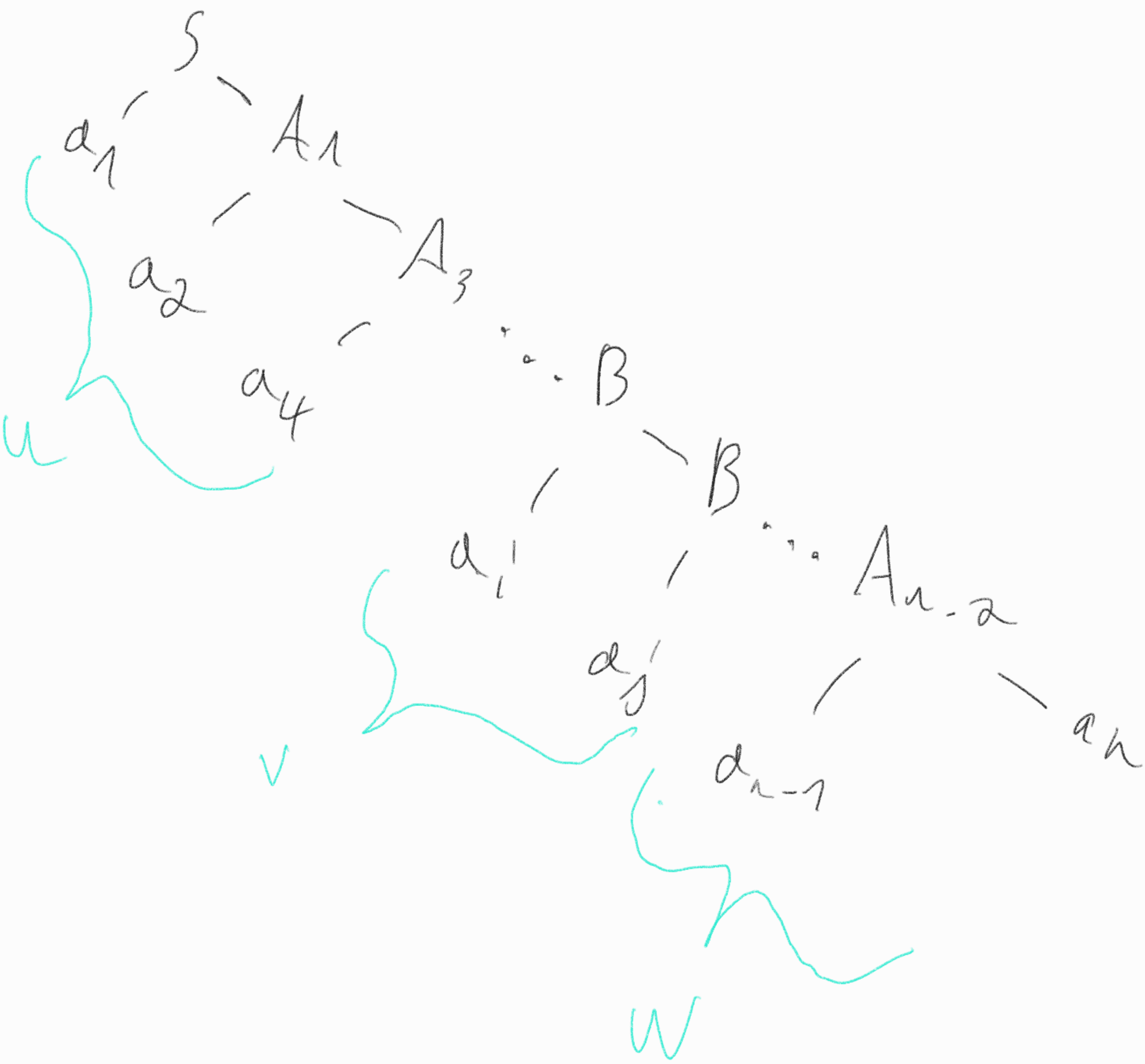
Ein Syntaxbaum von $x \in L$ sieht dann wie folgt aus:



Wir definieren n über $\#(N)+2$

Für jedes Wort $x \in L$ mit $|x| \geq n$ gilt:

Im Syntaxbaum kommen mind. $\#(N)+1$ Nichtterminale vor $\Rightarrow$ mindestens ein Nichtterminales kommt doppelt vor.



D.h. x kann in uvw zerlegt werden. mit $|v| \geq 1$, nach spätestens

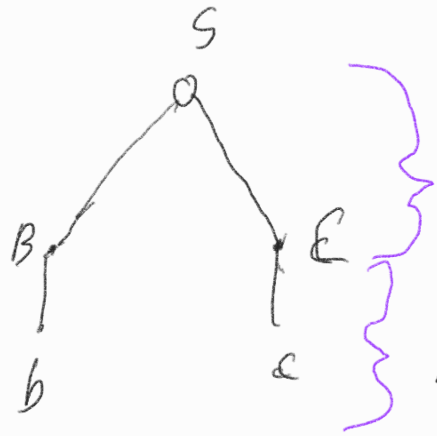
$\#(N)+1$ Nichtterminalen kommt

$\Rightarrow \underline{|uv| \leq} \#(N)+2 = \underline{n}$

und auch $\underline{uv^iw \in L}$.

8)

1) Für eine Chomsky-Normalform ergibt sich Syntaxbäume der Form.



Regel der Form $A \to BC$

Regel der Form $A \to a$

D.h. gilt $x \in L(G)$, dann sieht der Ersetzungsprozess wie folgt aus:

$S \vdash A_1 A_2 \vdash A_1 A_2' A_3' \vdash \dots \vdash A_1'' A_2'' \dots A_{|x|}''$

Da in jedem Ableitungsschritt genau ein Nichtterminales dazu kommt brauchen wir $|x|-1$ Schritte und $|x|$ um $A_1'' \dots A_{|x|}'' \vdash x$ zu erzeugen.

Zusammen: $2|x|-1$ Schritte

2) Gegeben: $G = (\{(,), \vee, \wedge, \neg, x\}, \{F\}, F, P)$
wobei $P = \{$

1. $F \to (F \wedge F)$
2. $F \to (F \vee F)$
3. $F \to \neg F$
4. $F \to x$

$\}$

ist nicht in Chomsky Normalform $\Rightarrow$

Konstruiere äquivalente Grammatik.

1. $F \to (F \wedge F)$ wird zu $F \to A F U F Z$
2. $F \to (F \vee F)$ — " — $F \to A F O F Z$
3. $F \to \neg F$ — " — $F \to N F$
4. Neue Regeln:

$A \to ($

$U \to \wedge$

$O \to \vee$

$Z \to)$

$N \to \neg$

Regeln kürzen:

- $F \to A R_1$, $R_1 \to F R_2$, $R_2 \to U R_3$, $R_3 \to F Z$, $R_2 \to O R_3$

$G' := (\{), (, 1, v, T, x\},$

$\{A, F, U, O, Z, N\}, F, P'\}$

= vor der Kürzung der Regeln

$G'' := (\{), (, 1, v, T, x\},$

$\{R_1, R_2, R_3, A, F, U, O, Z, N\}, F, P'\}$

= nach der Kürzung der Regeln