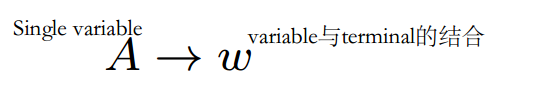
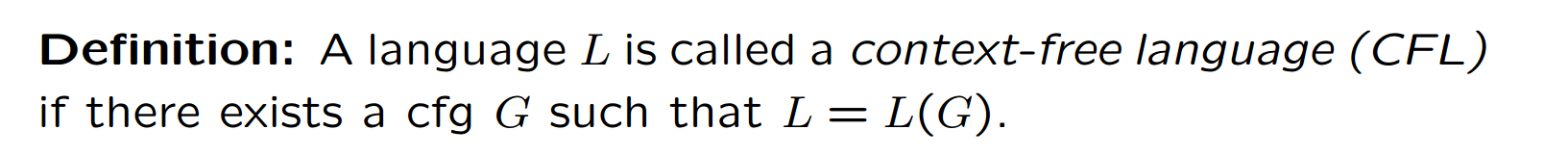
Context free grammar特点



w是大写字母小写字母的随机排列

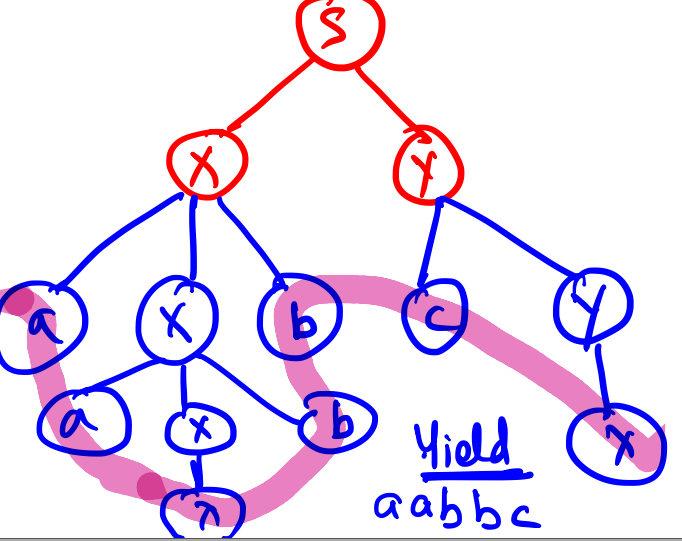
而regular grammar只能在一边而且最多只有一个大写字母



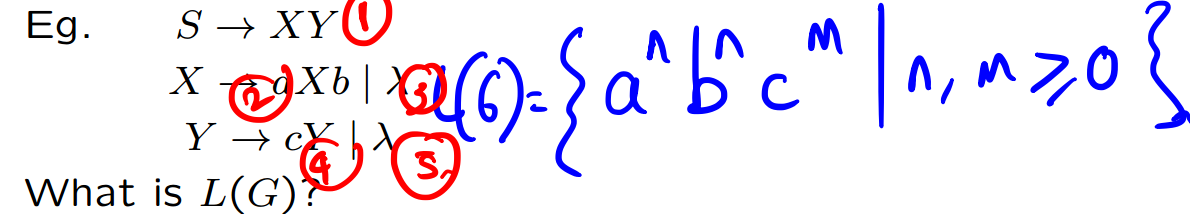
只要有regular grammar，他们组合的language就是cfl

regularLanguage也是cfl的一部分

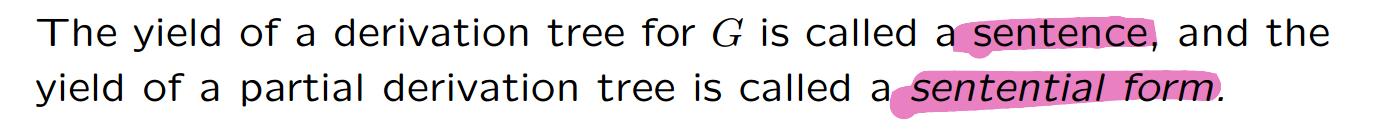
Derivation tree特点：root是S，Internal node是大写字母，leaf是小写字母，如果一个大写字母的儿子有一个入，那么就不能有其他字母



left most derivation就是优先化解X，right most derivation就是优先化解Y



Yield：读树最外面一圈总结出来的string



树的yield对G来说就是sentence，不完整的sentence叫做sentential form

Parsing：给你一个string和规则，让你倒推是怎么总结出这个string的

ambiguous:一个graammar被称作ambiguous如果他的其中一个string能被两个树表示

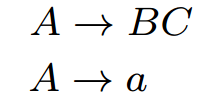
一个Language被称作unambiguous如果存在一个cfg不是ambiguous的

一个Language被称作inhenritly ambiguous如果每个grammar都是ambiguous的

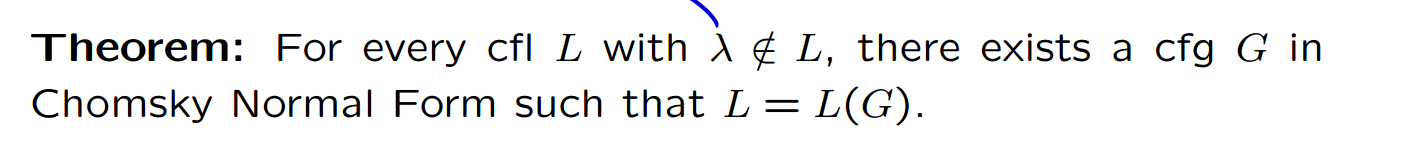
没有办法来决定grammar是不是ambiguous的

没有办法来决定language是不是inhenritly ambiguous的

CNF

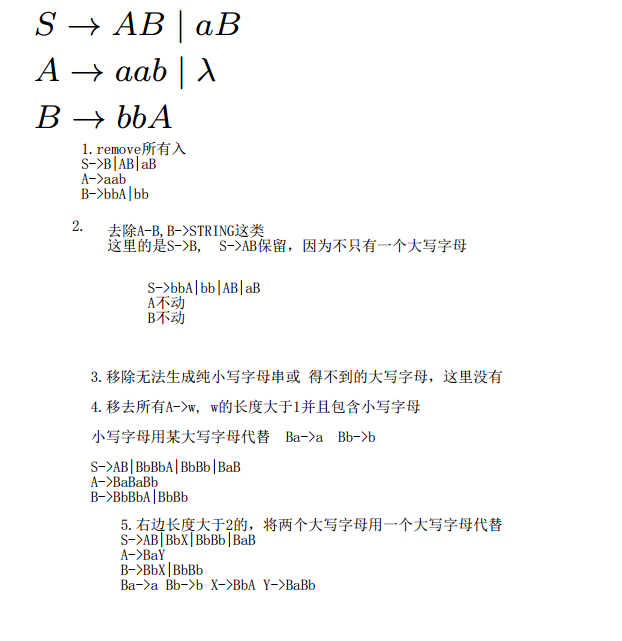


两个大写字母加一个小写字母

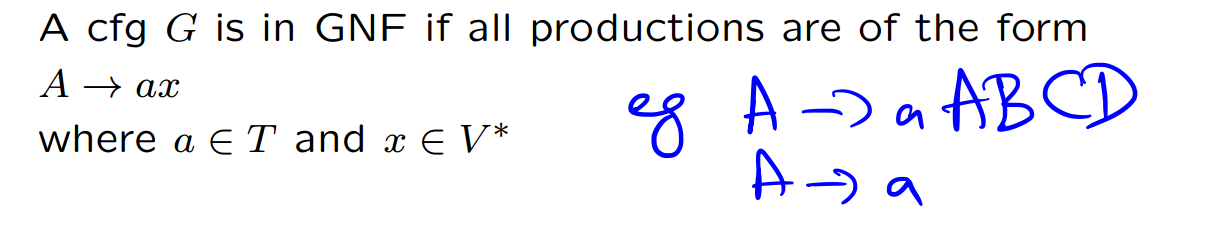


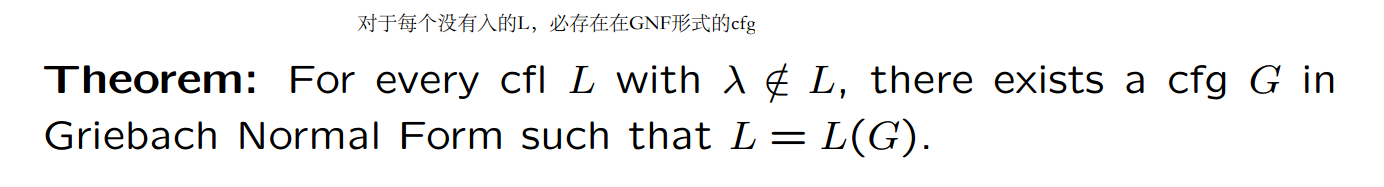
如果入不属于L，那就必然存在一个CNF来表示他

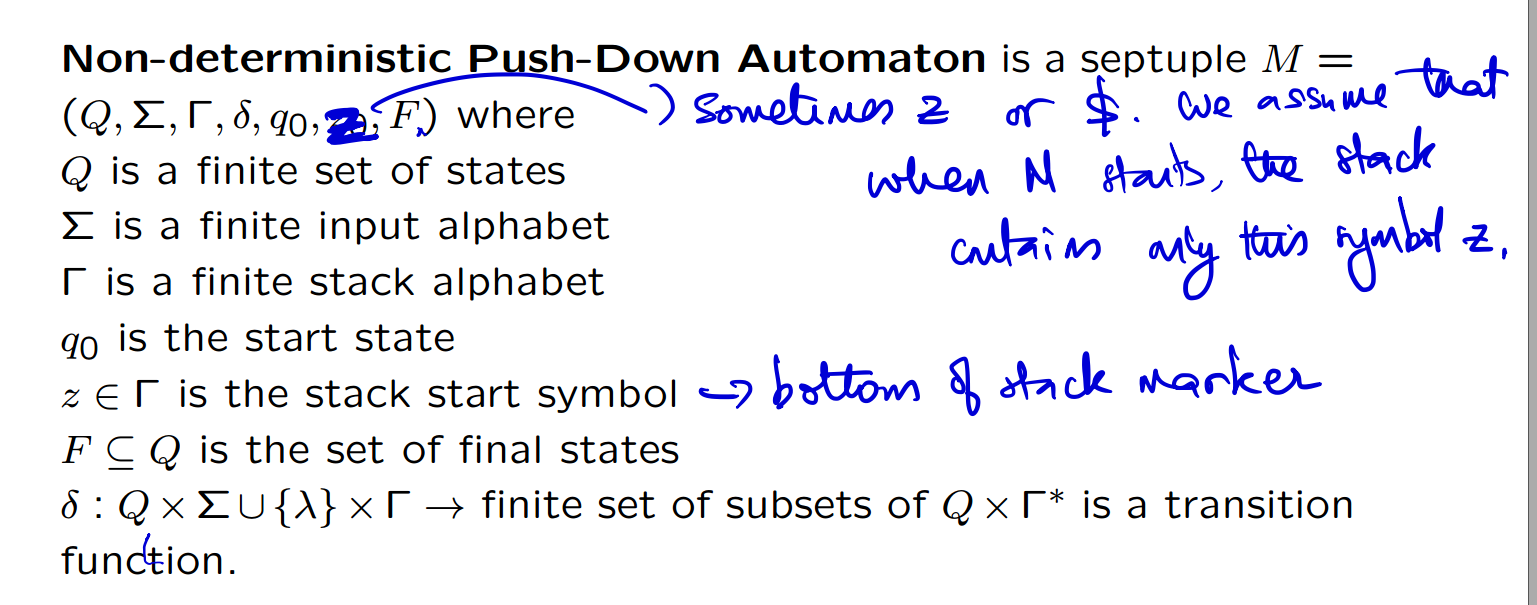
把CFG转换成CNF

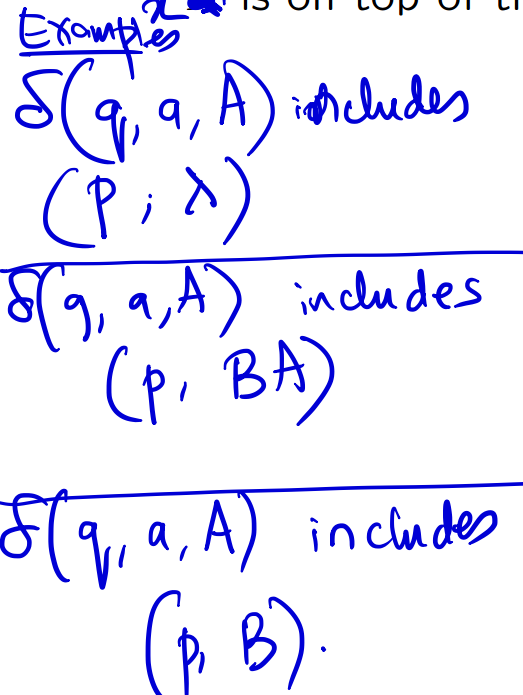


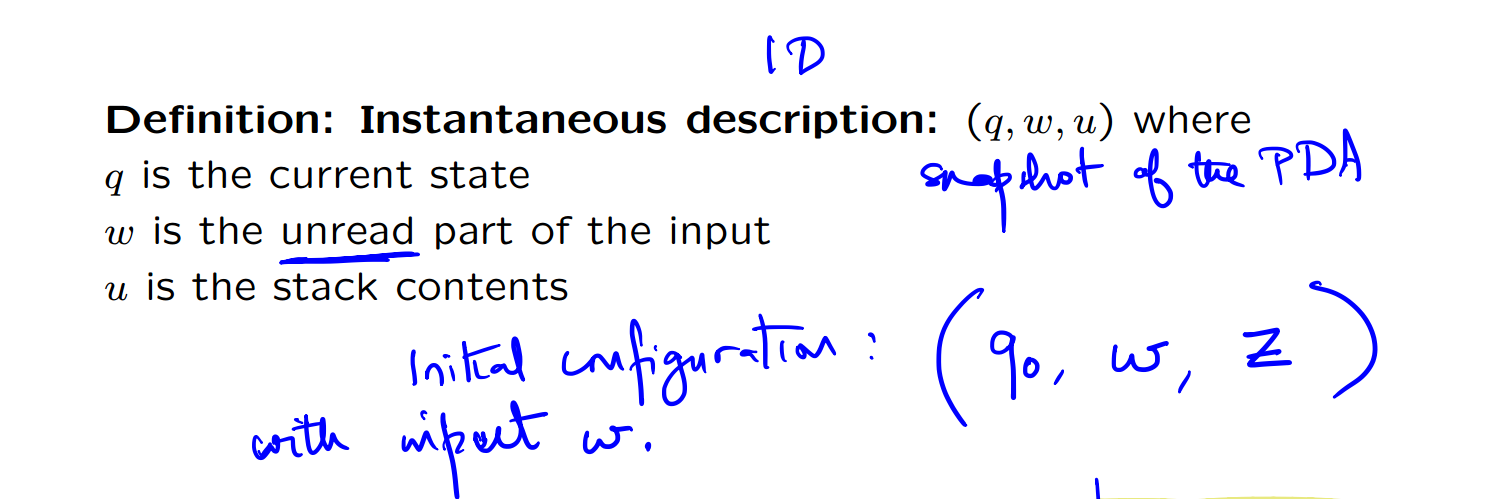
GNF



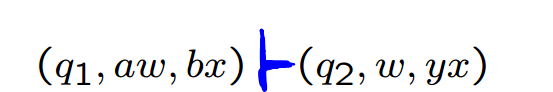


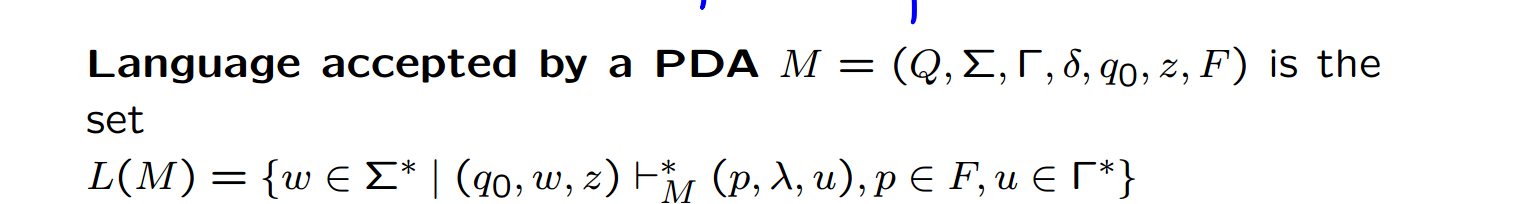




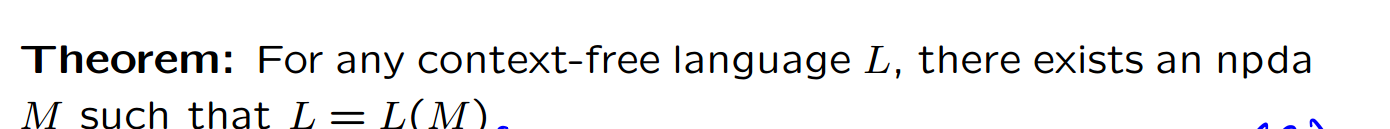
.

instantaneous descrpition不代表以后的状态，他只代表当前状态，这样才能有路径

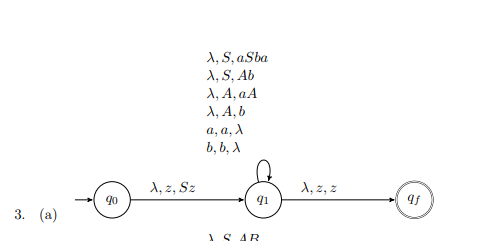




最后状态时p，z变成u



怎么把一个GRAMMAR转换成PDA



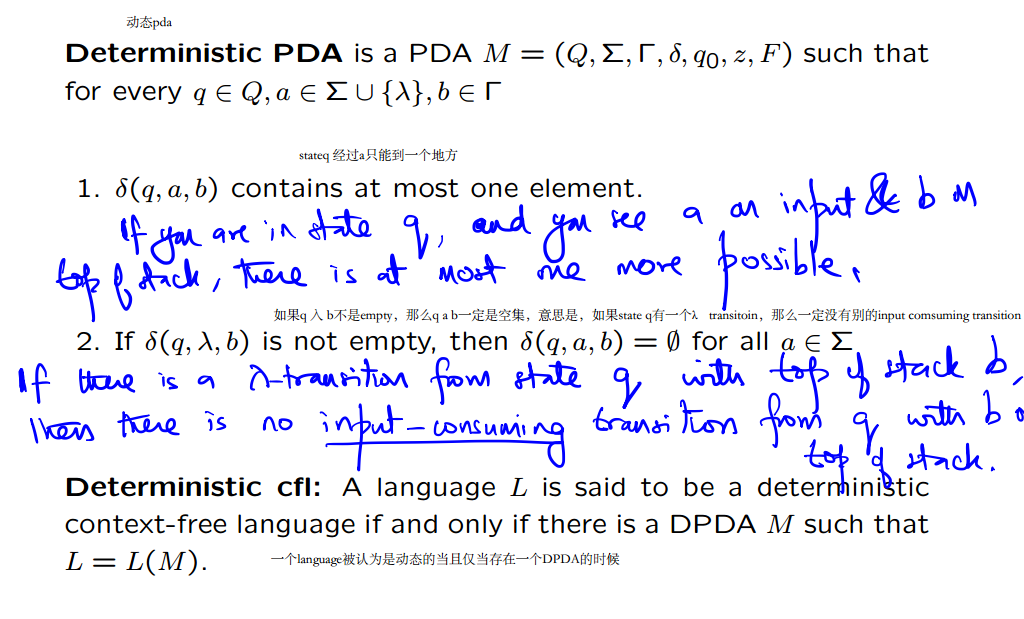
所有有的小写字母携程aa入

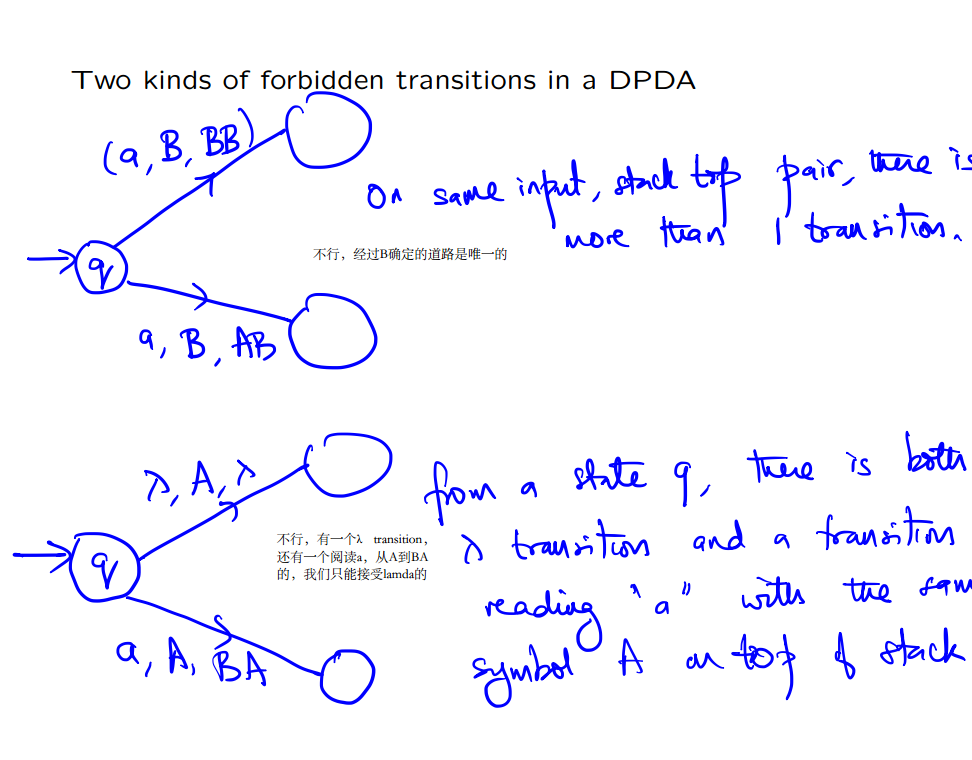
其他的全部转换成入，X,XXX

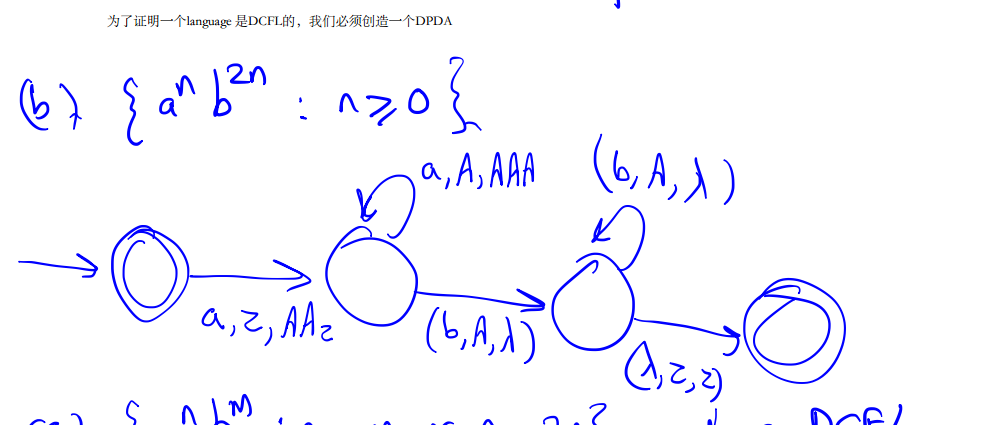
左边入z Sz

右边入zz

动态PDA

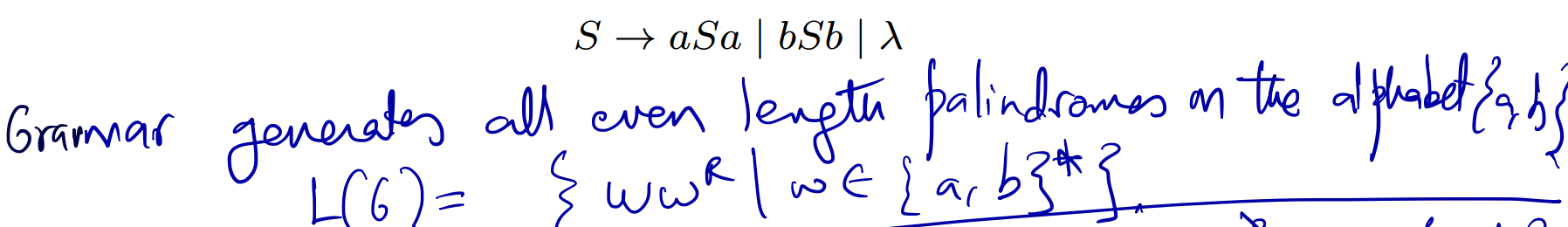


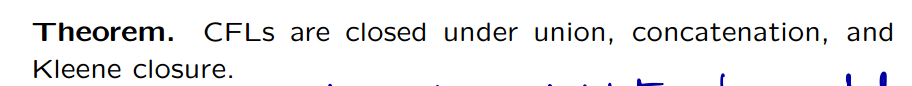


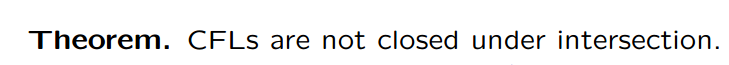
.

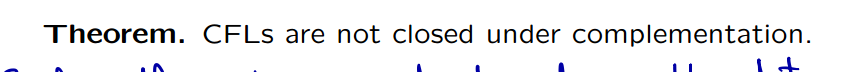
注意这个也是DPDA，在两个state之间有入就不能有别的，但是他自己本身循环还是可以的

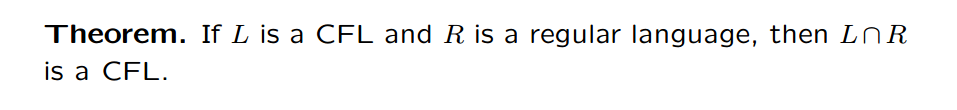
cflgrammar实例

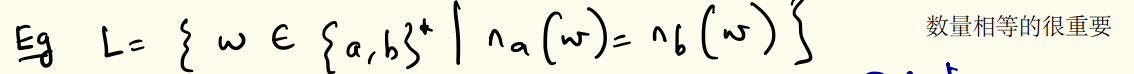


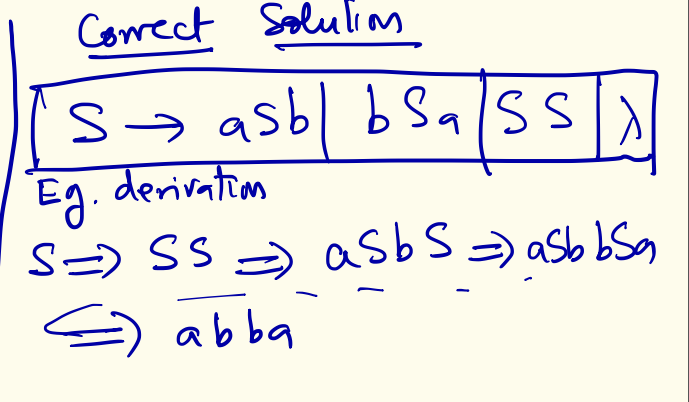


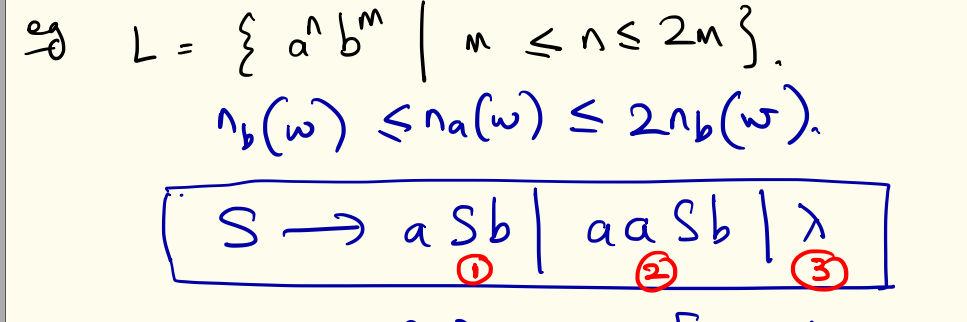




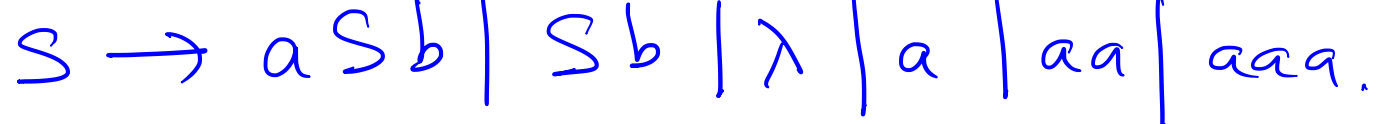


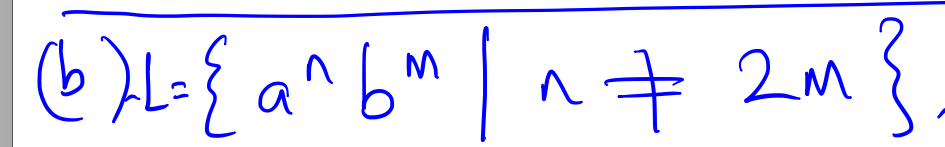


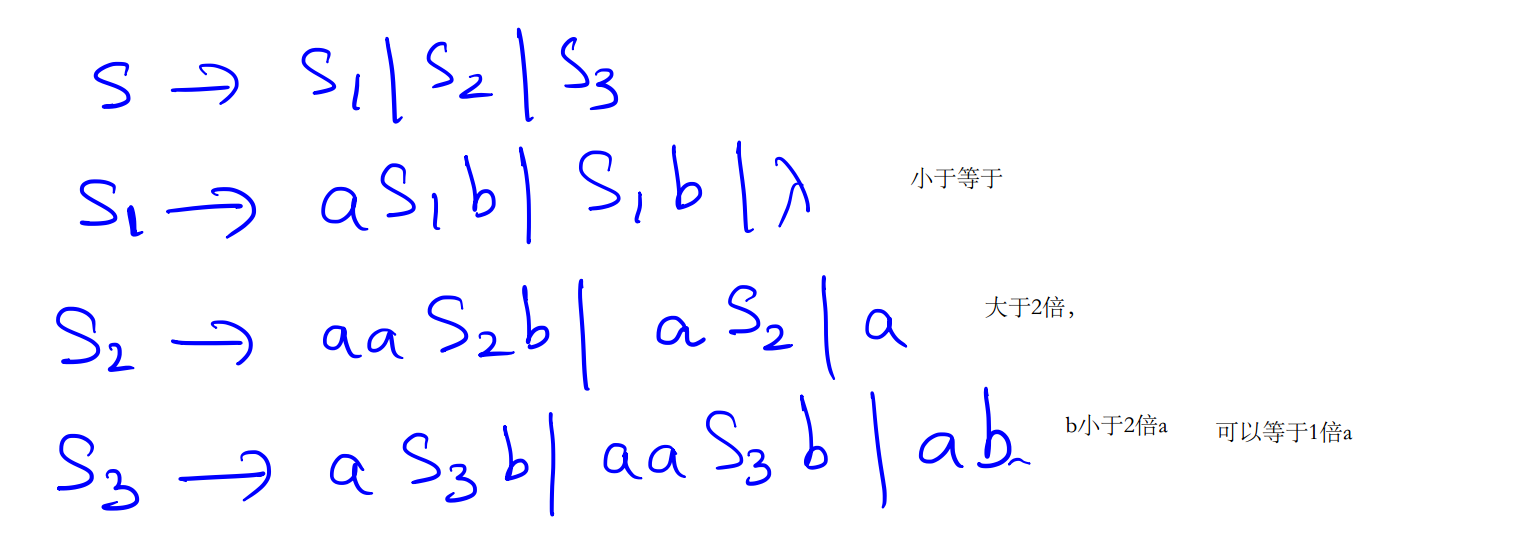




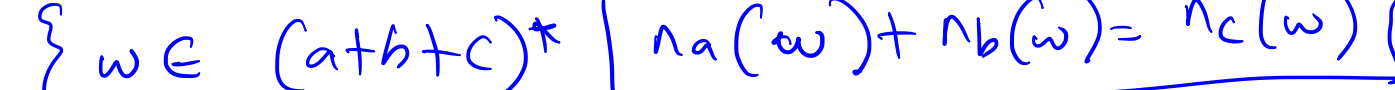


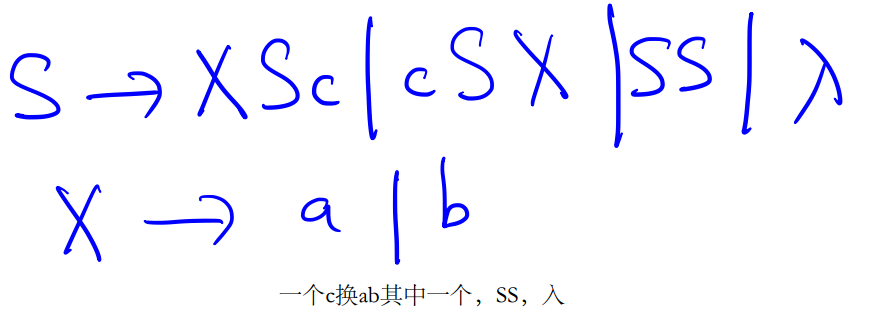


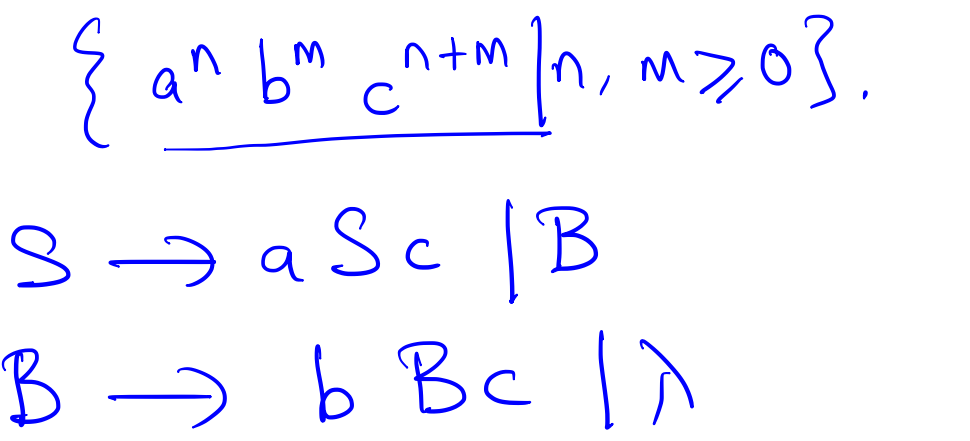


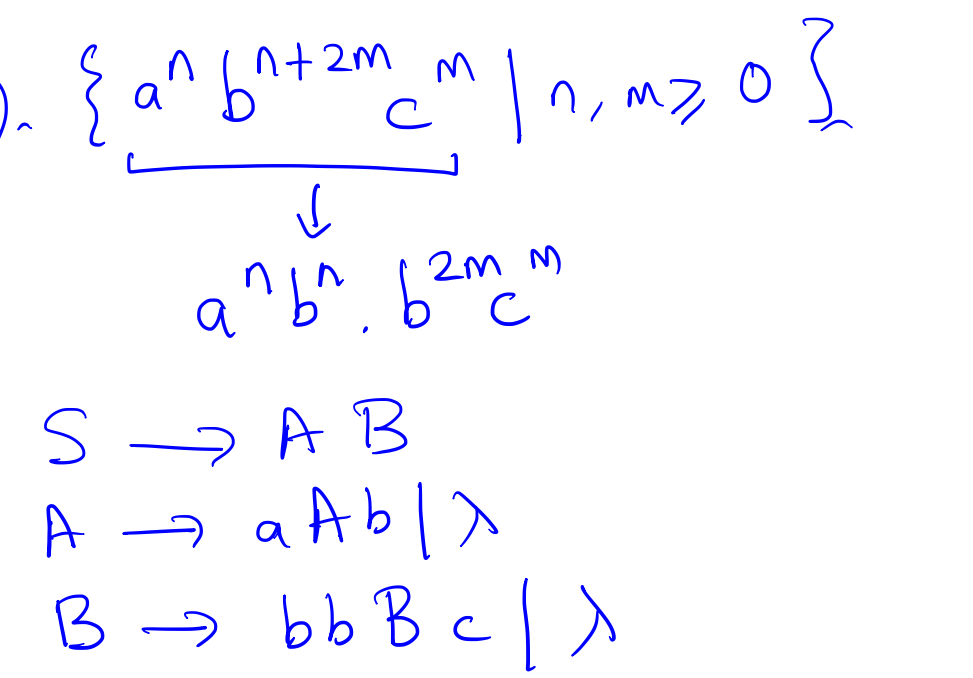


S3的ab要注意，他证明aS3b这个组合至少有一次，不然就有可能等于两倍m









复杂关系

