1. Construct an equivalent noncontracting grammar G_L with a nonrecursive start symbol. Give a regular expression.

G: $S \to aS \mid bS \mid B$
$B \to bb \mid C \mid \lambda$
$C \to cC \mid \lambda$

$\longrightarrow$

$S' \to S$
$S \to aS \mid bS \mid B$
$B \to bb \mid C \mid \lambda$
$C \to cC \mid \lambda$

Iteration	NULL	PREV
0	{B}	
1	{B, C}	{B}
2	{S, B, C}	{B, C}
3	{S', S, B, C}	{S, B, C}
4	{S', S, B, C}	{S', S, B, C}

$S' \to S \mid \lambda$
$S \to aS \mid bS \mid B \mid a \mid b$
$B \to bb \mid C$
$C \to cC \mid c$

$(a \cup b)^*(bb \cup \lambda)c^*$

$(bb \cup \lambda)$ covers one application of bb, or null, followed by 0 or more applications of C.

7. Construct an equivalent grammar G_C that does not contain chain rules. Give a regular expression.

G: $S \to AS \mid A$
$A \to aA \mid bB \mid C$
$B \to bB \mid b$
$C \to cC \mid B$

chain(S) = {S}
chain(A) = {A}
chain(B) = {B}
chain(C) = {C}

New	Prev	Chain(S)
{S}	∅	{S, A} chain S, A
{A}	{S, A}	{S, A, B, C} chain B, C
{B}	{S, A, B, C}	{S, A, B, C} no chain, B
{C}	{S, A, B, C}	{S, A, B, C} chain C, B

G_C $S \to AS \mid aA \mid bB \mid cC \mid bB \mid b$
$A \to aA \mid bB \mid b \mid cC \mid bB$
$B \to bB \mid b$
$C \to cC \mid bB \mid b$

14. Construct an equivalent grammar without using useless symbols. Trace the generation of sets TERM and REACH used to construct G_T and G_U

G: ✓$S \to AA \mid CD \mid bB$
✓$A \to aA \mid a$
$B \to bB \mid bC$
$C \to cB$
✓$D \to dD \mid d$

Term	Prev
{D, A}	∅
{S, D, A}	{D, A}
	{S, D, A}

$S \to AA$
$A \to aA \mid a$
$D \to dD \mid d$

Reach	Prev	New
{S}	∅	
{S, A}	{S}	{S}
{S, A}	{S, A}	{A}

$S \to AA$
$A \to aA \mid a$

23. $S \to A \mid ABa \mid AbA$
$A \to Aa \mid \lambda$
$B \to Bb \mid BC$
$C \to CB \mid CA \mid bB$

Remove λ rules

NULL	PREV
{A}	∅
{S, A}	{A}
{S, A}	{S, A}

$\longrightarrow$

$S \to A \mid ABa \mid AbA \mid Ba \mid bA \mid Ab \mid \lambda$
$A \to Aa \mid a$
$B \to Bb \mid BC$
$C \to CB \mid CA \mid bB$

23 (cont.)

Eliminate Chain rules

$S \to A \mid ABa \mid AbA \mid Ba \mid bA \mid Ab \mid \lambda$
$A \to Aa \mid a$
$B \to Bb \mid BC$
$C \to CB \mid CA \mid bB$

Chain(S)	PREV	NEW
{S}	∅	
{S}	{S}	{S}
{S, A}	{S, A}	{A}

Thus, add rules from A to S

$S \to A \mid ABa \mid AbA \mid Ba \mid bA \mid Ab \mid \lambda \mid Aa \mid a$
$A \to Aa \mid a$
$B \to Bb \mid BC$
$C \to CB \mid CA \mid bB$

Useless symbols

Term	Prev
{A}	
{S, A}	{A}
{S, A}	{S, A}

$S \to Aa \mid a \mid AbA \mid bA \mid Ab \mid b \mid \lambda$
$A \to Aa \mid a$

Reach	Prev	New
{S}	∅	
{S, A}	{S}	{S}
{S, A}	{S, A}	{A}

Chomsky transformation

$S \to AA' \mid a \mid AT_1 \mid B'A \mid AB' \mid b \mid \lambda$
$A \to AA' \mid a$
$T_1 \to B'A$
$A' \to a$
$B' \to b$

26. Use the CYK algorithm to produce the upper diagonal matrix of the chomsky grammar:

$S \to AX \mid AY \mid a$
$X \to AX \mid a$
$Y \to BY \mid a$
$A \to a$
$B \to b$

	1	2	3	4	5
1	{S, X, Y, A}	∅	{S}	∅	∅
2		{B}	{Y}	∅	∅
3			{S, X, Y, A}	{S, X}	{S, X}
4				{A, Y, X, S}	{S, X}
5					{A, Y, X, S}