You have one hour and 15 minutes to complete the exam. Do not open the exam until instructed to do so. No notes, texts, computers, calculators, or communication devices are permitted. Write all answers on the examination paper itself. BUDGET YOUR TIME WELL! SHOW ALL WORK!

Grades out of 50 are: 12 15 16 16 20 22 24 26 26 28 28 29 29 30 30 31 33 33 35 36 36 37 37 37 37 37 38 38 39 39 39 40 40 41 42 42 42

Bonus question [1 mark]: What is $\emptyset^+?$  It is $\emptyset.$

Question 1. [12 marks total]

(a) [2 marks] What is a normal form for a grammar?
A normal form (1) imposes some restriction on the format of the rules of the grammar, but (2) does not change the set of languages generated.

(b) [2 marks] Give a formal definition of context-free grammars.
A context-free grammar is a 4-tuple $(V, \Sigma, P, S)$ where $V$ is a finite set of variables, $S$ is a variable in $V$ called the start variable, $\Sigma$ is a finite set of terminals, and $P$ is a set of rules; each rule is of the form $A \rightarrow w$ where $A$ is in $V$ and $w$ is in $(V \cup \Sigma)^*$.

(c) [3 marks] Give a formal definition of Chomsky normal form for context-free grammars.
A CFG is in CNF if every rule is of the form $A \rightarrow BC$, $A \rightarrow a$, or $S \rightarrow \lambda$, where $S$ is the start variable, $A$ is any variable, $B$ and $C$ are any variable except the start variable, and $a$ is any terminal.

(d) [5 marks] Give a Chomsky normal form grammar for $ww^{rev}$ with $w$ in $\{a, b\}^*$,
where $w^{rev}$ is the reverse of $w.$

$S \rightarrow \lambda | AT | BR$
$M \rightarrow AT | BR$
$T \rightarrow a | MA$
$R \rightarrow b | MB$
$A \rightarrow a$
$B \rightarrow b$
NOTE: You cannot have an S on the right hand side of a rule!!!

Question 2. [12 marks total]
(a) [4 marks] Give a context-free grammar for the language \( \{a^n b^n a^m b^m : n \geq 0, m \geq 2\} \).
\[
S \to aSb | bMa \\
M \to bMa | \lambda
\]
(b) [4 marks] Give a context-free grammar for the language of palindromes over the alphabet \{a,b,c\}.
\[
S \to aSa | bSb | cSc | a | b | c | \lambda
\]
(c) [4 marks] Give a context-free grammar for the language of strings over alphabet \{a,b\} that contain at least as many a’s as b’s.
\[
S \to SaSb | SbSa | SS | a | \lambda
\]

Question 3 [16 marks] Let \( G \) be the grammar with rules
\[
S \to aABb | A | B \\
A \to aSSb | C | D | \lambda \\
B \to ab | BC | ECE \\
C \to aABb | CD | DC | \lambda \\
D \to E | DE | EF \\
E \to DD \\
F \to BB | \lambda
\]
(a) [7 marks] Which variables of \( G \) are nullable? useless? recursive?

<table>
<thead>
<tr>
<th>Variable</th>
<th>Nullable?</th>
<th>Yes/No</th>
<th>Useless?</th>
<th>Yes/No</th>
<th>Recursive?</th>
<th>Yes/No</th>
</tr>
</thead>
<tbody>
<tr>
<td>S</td>
<td>Yes</td>
<td></td>
<td>No</td>
<td></td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>A</td>
<td>Yes</td>
<td></td>
<td>No</td>
<td></td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>B</td>
<td>No</td>
<td></td>
<td>No</td>
<td></td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>Yes</td>
<td></td>
<td>No</td>
<td></td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>D</td>
<td>No</td>
<td></td>
<td>Yes</td>
<td></td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>E</td>
<td>No</td>
<td></td>
<td>Yes</td>
<td></td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>F</td>
<td>Yes</td>
<td></td>
<td>Yes</td>
<td></td>
<td>Yes</td>
<td></td>
</tr>
</tbody>
</table>

(b) [3 marks] Determine the chain for each variable of \( G \).

<table>
<thead>
<tr>
<th>Variable</th>
<th>Chain of the variable</th>
</tr>
</thead>
<tbody>
<tr>
<td>S</td>
<td>{S, A, B, C, D, E}</td>
</tr>
<tr>
<td>A</td>
<td>{A, C, D, E}</td>
</tr>
<tr>
<td>B</td>
<td>{B}</td>
</tr>
</tbody>
</table>
(c) [6 marks] Using your results from the earlier parts, modify the grammar so that (1) the start variable is not recursive, (2) there are no $\lambda$-rules except possibly $S \rightarrow \lambda$, and (3) there are no chain rules.

First eliminate useless variables and their rules, and make the start variable not recursive:

\[
\begin{align*}
S & \rightarrow T \\
T & \rightarrow aABb \mid A \mid B \\
A & \rightarrow aTTb \mid C \mid \lambda \\
B & \rightarrow ab \mid BC \\
C & \rightarrow aBAAb \mid \lambda
\end{align*}
\]

Eliminate $\lambda$-rules and simplify:

\[
\begin{align*}
S & \rightarrow T \mid \lambda \\
T & \rightarrow aABb \mid aBb \mid A \mid B \\
A & \rightarrow aTTb \mid aTb \mid ab \mid aBAAb \mid aBb \\
B & \rightarrow ab \mid BaBAAb \mid BaBb
\end{align*}
\]

Eliminate chain rules:

\[
\begin{align*}
S & \rightarrow aABb \mid aBb \mid aTTb \mid aTb \mid ab \mid aBAAb \mid BaBAAb \mid BaBb \mid \lambda \\
T & \rightarrow aABb \mid aBb \mid aTTb \mid aTb \mid ab \mid aBAAb \mid BaBAAb \mid BaBb \\
A & \rightarrow aTTb \mid aTb \mid ab \mid aBAAb \mid aBb \\
B & \rightarrow ab \mid BaBAAb \mid BaBb
\end{align*}
\]

Question 4 [10 marks]

(a) [3 marks] Give a formal definition for a pushdown automaton (PDA).

Sudkamp Definition 7.1.1.

(b) [2 marks] State precisely when a pushdown automaton is deterministic.

Sudkamp page 225.

(c) [5 marks] Draw a state diagram of a pushdown automaton that accepts exactly the palindromes over \{a,b\} of odd length, where acceptance must be by final state and empty stack.

Take Example 7.1.3 of Sudkamp and change the middle transition to two transitions, one labeled “a $\lambda/\lambda$” and the other labeled “b $\lambda/\lambda$”.