

1. Write a **Context Free Grammar** for the language

$$L = \{ a^k b^m c^n \mid k = n + m, \text{ or } m = k + n, \text{ or } n = k + m, k > 0, m > 0, n > 0 \}.$$

2. Consider the language

$$L = \{ a^n b^{n!} \mid n > 0 \}.$$

Use the Pumping Lemma for Context-Free Languages to show that **L** is not context-free.

3. Present the **CKY** recognition matrix for the string **bbabb** assuming the Chomsky Normal Form grammar, $G = (\{S, A, B, C, D\}, \{a, b\}, R, S)$, specified by the rules **R**:

$$S \rightarrow AB \mid BA \mid BD$$

$$A \rightarrow CS \mid CD \mid a$$

$$B \rightarrow DS \mid b$$

$$C \rightarrow a$$

$$D \rightarrow b$$

		b	b	a	b	b
1						
2						
3						
4						
5						

4. Choosing from among **(Y) yes**, **(N) No**, categorize each of the following closure properties. No proofs are required.

Problem / Language Class (C)	Regular	Context Free
Closed under union with Context Free languages?		
Closed under quotient with languages of its own class (C), i.e., $L1/L2$		
Closed under difference with languages of its own class (C), i.e., (difference $(L1, L2) = L1 - L2$)?		
Closed under intersection with Context Free languages?		

5. Prove that any class of languages, C , closed under union, concatenation, intersection with regular languages, homomorphism and substitution (e.g., the Context-Free Languages) is closed under **Erase Middle with Regular Sets (em)**, where $L \in C$, R is Regular, L and R are over the alphabet Σ , and $L \text{ em } R = \{ xz \mid x, z \in \Sigma^+ \text{ and } \exists y \in R, \text{ such that } xyz \in L \}$. You may assume substitution $f(a) = \{a, a'\}$, and homomorphisms $g(a) = a'$ and $h(a) = a, h(a') = \lambda$. Here $a \in \Sigma$ and a' is a distinct new character associated with each $a \in \Sigma$.
You must be very explicit, describing what is produced by each transformation you apply.

6. Consider the CFG $G = (\{S, T\}, \{a, b\}, R, S)$ where R is:

$S \rightarrow a T T \mid T S \mid a$

$T \rightarrow b S T \mid b$

a.) Present a pushdown automaton that accepts the language generated by this grammar. You may (and are encouraged) to use a transition diagram where transitions have arcs with labels of form $a, \alpha \rightarrow \beta$ where $a \in \Sigma \cup \{\lambda\}$, $\alpha, \beta \in \Gamma^*$. Note: I am encouraging you to use extended stack operations.

What parsing technique are you using?

(Circle one) **top-down** or **bottom-up**

How does your PDA accept? (Circle one) **final state** or **empty stack** or **final state and empty stack**

What is the **initial state**? _____

What is the **initial stack content**? _____

What are your **final states** (if any)? _____

What parsing technique are you using? (Circle one) **top-down** or **bottom-up**

How does your PDA accept? (Circle one) **final state** or **empty stack** or **final state and empty stack**

What is the **initial state**? _____

What is the **initial stack content**? _____

What are your **final states** (if any)? _____

b.) Now, using the notation of **IDs** (Instantaneous Descriptions, $[q, x, z]$), describe how your PDA accepts strings generated by G .

7. Consider the context-free grammar $G = (\{ S, A, B \}, \{ a, b \}, R, S)$, where R is:

$$S \rightarrow SAB \mid BA$$

$$A \rightarrow AB \mid a$$

$$B \rightarrow bS \mid b \mid \lambda$$

a.) Remove all λ -rules from G , creating an equivalent grammar G' . Show all rules.

$$\text{Nullable} = \{ \quad \}$$

G' :

b.) Remove all **unit** rules from G' , creating an equivalent grammar G'' . Show all rules.

$$\text{Unit}(S) = \text{Chain}(S) = \{ \quad \}; \text{Unit}(A) = \{ \quad \}; \text{Unit}(B) = \{ \quad \}$$

G'' :

c.) Convert grammar G'' to its **Chomsky Normal Form** equivalent, G''' . Show all rules.

G''' :

In exam I may have some Unproductive non-terminals and some Unreachable ones.