COT 4210 Fall 2016 **Midterm 2** **Name: ________________**
Total Points Available 60 **Your Raw Score _______** **Grade:____________**

2 **1.** Let $\mathbf{A} = (\{q1, \ldots q10\}, \{0,1\}, q1, \{q3,q7\})$ be some DFA. Assume you have computed the sets, $\mathbf{R^k_{i,j}}$, for $\mathbf{0 \le k \le 10, 1 \le i \le n, 1, 1 \le j \le n}$. In terms of the $\mathbf{R^k_{i,j}}$, what is the language recognized by **A**?

$L = R^{10}_{1,3} + R^{10}_{1,7}$

4 **2.** Write a Context Free Grammar for the language **L**, where
L = { w | w ∈ {a,b}* and w has twice as many a's as b's }.
Hint: You need to consider all possible relative orderings of **a**'s and **b**'s

$S \rightarrow S\,a\,S\,a\,S\,b\,S \mid S\,a\,S\,b\,S\,a\,S \mid S\,b\,S\,a\,S\,a\,S \mid \lambda$

5 **3.** Use Myhill-Nerode to show that the following language **L** is not Regular.
$\mathbf{L = \{ a^k\ b^{k^2} \mid k > 0 \}}$.
Hint: Use the right-invariant equivalence relation $\mathbf{R_L}$, where $\mathbf{x\ R_L\ y}$ **iff** $\mathbf{\forall z[xz \in L \Leftrightarrow yz \in L]}$

Consider the equivalence classes $[a^n]_{R_L}$, $n > 0$.

The equivalence class $[a^i]_{R_L}$ is such that $a^i b^{i^2} \in L$

However, the equivalence class $[a^j]_{R_L}$ is such that $a^j b^{i^2} \in L$ iff $i = j$.

Thus, each $[a^i]_{R_L}$ is a unique class and so there are an infinite number of such classes showing that L is not a Regular language.

7 **4.** Consider the language

L = { $a^n b^n c^m$ | m > n }.

Let N be chosen by the P.L.
Choose $w = a^N b^N c^{N+1} = uvwxy$, $|vwx| \le N$, $|v| + |x| > 0$.

Case 1) vwx is over a's or b's or both but no c's, choose i = 2:
uv^2xy^2z has at least N+1 a's or N+1 b's or at least N+1 of each. In all three cases, there are not sufficient c's to be greater than both the a's and b's

Case 2) vwx is over c's or c's and b's. Set i=0 and we will have at least one fewer c's and so there will be at least as many a's as c's.

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

S → AB | BA
A → CX | a
B → XD | b
C → XA
D → BX
X → a | b

	a	b	b	a	a	b
1	*A,X*	*B,X*	*B,X*	*A,X*	*A,X*	*B,X*
2	*S*	*D*	*S,D,C*	*C*	*S*	
3	*B*	*B*	*A*	*A*		
4	*S,D*	*S,C,D*	*S,C*			
5	*B*	*A*				
6	*C,S,D*					

Is **abbaab** in *L*(G)?_____***YES***___

How do you know from above?_***S appears in last cell of CKY matrix***_______________

4 **6**. Give an explicit example of two Context Free Languages, L1 and L2, whose intersection is the non-Context Free Language { $\mathbf{a^n\ b^n\ c^n \mid n \geq 0}$ }. No grammars or proof is required.

L1 = _____ $\{a^n\ b^n\ c^m \mid n, m \geq 0\}$ ________________________

L2 = _____ $\{a^m\ b^n\ c^n \mid n, m \geq 0\}$ ________________________

Give an explicit example of a Regular Language, **R**, whose intersection with the Context Sensitive **L =** { $\mathbf{a^n\ b^n\ c^n \cup a^n\ b^n \mid n \geq 0}$ } is a Context Free, non-Regular Language. No grammars or proof is required, but you must describe the language produced by this intersection

R = _____ $a^*\ b^*$ ________________________

L ∩ R = _ $\{\ a^n\ b^n \mid n \geq 0\ \}$ ________________________

8 **7**. 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 **huh** where $\mathbf{L} \in \boldsymbol{C}$, **R** is Regular, **L** and **R** are over the alphabet **Σ**, and

L huh R = $\{ \mathbf{y} \mid \mathbf{y} \in \Sigma^+$ and $\exists\ \mathbf{x,z} \in \mathbf{R}^+$, such that $\mathbf{xyz} \in \mathbf{L} \}$.

You may assume substitution **f(a) = {a, a'},** and homomorphisms **g(a) = a'** and **h(a) = a, h(a') = λ**. Here $\mathbf{a} \in \Sigma$ and **a'** is a distinct new character associated with each $\mathbf{a} \in \Sigma$.

You must be very explicit, describing what is produced by each transformation you apply and what kind of language results.

L huh R = h(f(L) ∩ (g(R⁺) Σ⁺ g(R⁺)))

f(L) = { <u>w</u> | w ∈ L } where <u>w</u> has some (or none) of its letters primed. f(L) is a CFL since CFLs are closed under substitution.

g(R⁺) = { y' | y ∈ R⁺ } where y' has all of its letter primed. g(R⁺) is Regular since Regular languages are closed under Kleene + and homomorphism.

g(R⁺) Σ⁺ g(R⁺) = { xy'z | x,z R⁺ and y ∈ Σ⁺, This is a Regular language since Regular languages are closed under concatenation.

f(L) ∩ (g(R⁺) Σ⁺ g(R⁺)) = { x'yz' | xyz ∈ L and x,z ∈ R⁺ } . This is a CFL since CFLs are closed under intersection with Regular.

L huh R = h(f(L) ∩ (g(R⁺) Σ⁺ g(R⁺))) = { y | ∃x,z ∈ R⁺ where xyz ∈ L } is a CFL since CFLs are closed under homomorphism.

8 **8.** Consider the CFG **G = ({ S, T, C, D } , { a , b, c, d } , R , S)** where **R** is:

S → a T S | a T D
T → b T S | b C
C → c
D → d

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, α → β** where **a** ∈ Σ∪{λ}, α, β ∈ Γ*. Note: You are allowed to use extended stack operations that push more than one symbol onto stack.

Bottom Up (BU)
In state q (starting state)
Shift:
a,λ / a; b,λ / b; c,λ / c; d,λ / d
In state q (starting state)
Reduce:
λ,aTS / S; λ,aTD / S; λ,bTS / T; λ,bC / T; λ,c / C; λ,d / D
Accept:
λ,S$ / λ (and enter state f)

OR Since in GNF can do
In state q (starting state)
a,TS / S; a,TD / S; b,TS / T; b,C / T; c,λ / C; d,λ / D
Accept:
λ,S$ / λ (and enter state f)

Top Down (TD)
In state q (starting and only state)
a,a / λ; b,b / λ; c,c / λ; d,d / λ
λ,S / aTS; λ,S / aTD; λ, T / bTS; λ, T / bC; λ,C / c; λ,D / d

OR Since in GNF can do
In state q (starting and only state)
a,S / TS; a,S / TD; b,T / TS; b,T / C ; c,C / λ; d,D / λ

b.) 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**? *BU: q ; TD: q*
What is the **initial stack content**? *BU: $; TU: S*
What are your **final states** (if any)? *BU: f; TU: N/A*

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

Bottom Up [q, w, $] |--* [f, λ, λ]; Top Down [q, w, S] |--* [q, λ, λ]

2 **9.** Consider the context-free language **L = { $a^n b^m$ | n<m }**. What language results when we take the **Max** of this language? What about the **Min**? To help you recall definitions, here they are.

Max(L) = { w | w $\in$ **L**, and if **wy** $\in$ **L**, then **y = λ }**
Max says that a string is kept only if that string is not a proper prefix of another string in L
Give your explicit answer for L = { $a^n b^m$ | n<m } below
Max(L) = $\varnothing$

Min(L) = { w | w $\in$ **L**, and if **xy = w** and **x** $\in$ **L**, then **y = λ }**
Min says that a string is kept only if no proper prefix of that string is in L
Give your explicit answer for L = { $a^n b^m$ | n<m } below
Min(L) = { $a^n b^{n+1}$ | $n \geq 0$ }

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

S → aAa | bBb
A → CA | AB
B → C | b
C → D | λ
D → abC

3 **a.)** Remove λ-rules from **G**, creating an equivalent grammar **G**'. Show all rules.
Nullable = { B, C, D }
S → aAa | bBb | bb
A → CA | AB | A ***// can omit the A or not***
B → C | b
C → D
D → abC | ab

2 **b.)** Remove all **unit** rules from **G'**, creating an equivalent grammar **G**''. Show all rules.
Chain(S) = { S }; Chain(A) = { A }; Chain(B) = { B, C, D};
Chain (C) = { C, D}; Chain(D) = { D }
S → aAa | bBb | bb
A → CA | AB
B → b | abC | ab
C → abC | ab
D → abC | ab

2 **c.)** Remove all useless symbols, creating an equivalent grammar **G'''**. Show all rules.
Unproductive = {A }; Unreachable = { D }
S → bBb | bb
B → b | ab | abC
C → abC | ab

3 **d.)** Convert grammar **G'''** to its **Chomsky Normal Form** equivalent, $\mathbf{G^{iv}}$. Show all rules.

S → <bB><b> | <b><b>
<bB> → <b>B
B → b | <a><b> | <ab>C
C → <ab>C | <a><b>
<ab> → ab
<a> → a
<b> → b