Middle East Technical University Department of Computer Engineering

Formal Languages and Abstract Machines
Take Home Exam 2

Alper KOCAMAN
2169589

1 Context-Free Grammars (10 pts)

a) Give the rules of the Context-Free Grammars to recognize strings in the given languages where $\Sigma = \{a, b\}$ and S is the start symbol.

$L(G) = \{w \mid w \in \Sigma^*;\ |w| \geq 3;$ the first and the second from the last symbols of w are the same$\}$ (2/10 pts)

$S \rightarrow aMab|aMaa|bMba|bMbb$

$M \rightarrow aM|bM|MM|\varepsilon$

$L(G) = \{w \mid w \in \Sigma^*;$ the length of w is odd$\}$ (2/10 pts)

$S \rightarrow aM|bM$

$M \rightarrow aMa|aMb|bMa|bMb|\varepsilon$

$L(G) = \{w \mid w \in \Sigma^*;\ n(w,a) = 2 \cdot n(w,b)\}$ where $n(w,x)$ is the number of x symbols in w (3/10 pts)

$S \rightarrow aSaSbS|aSbSaS|bSaSaS|\varepsilon$

b) Find the set of strings recognized by the CFG rules given below: (3/10 pts)

$S \to X \mid Y$
$X \to aXb \mid A \mid B$
$A \to aA \mid a$
$B \to Bb \mid b$
$Y \to CbaC$
$C \to CC \mid a \mid b \mid \varepsilon$

S can select X or Y.
If X is chosen,then recognized string in that form:
First 0 or more a's are in the string,then 0 or more b's come but number of a's and number of b's cannot be equal.
Formally,let the set of recognized string be X,than set $X = \{(a^*b^*)\backslash(a^nb^n) : n \geq 1\} \cup \{a\} \cup \{b\}$ or $X = \{(a^nb^m) : n \geq 0, m \geq 0, n \neq m\}$.

Y can also be selected.
If Y is chosen,then recognized string in that form:
The string contains at least 1 'b' and after that 'b',at least 1 'a',namely this string contains 'ba' as a substring.
Formally,let the set of recognized string be Y,than set $Y = \{(a^*b^*)(ba)(a^*b^*)\}$.

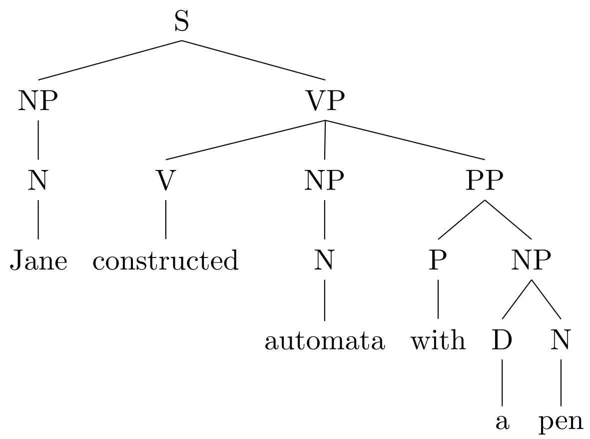
Thus,the recognized set of strings are $X \cup Y$.
$X \cup Y = \{(a^nb^m) : n \geq 0, m \geq 0, n \neq m\} \cup \{(a^*b^*)(ba)(a^*b^*)\}$

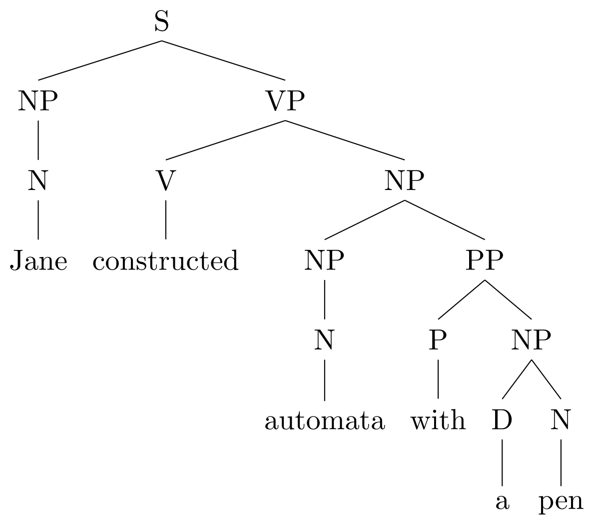
2 Parse Trees and Derivations (20 pts)

Given the CFG below, provide parse trees for given sentences in **a** and **b**.

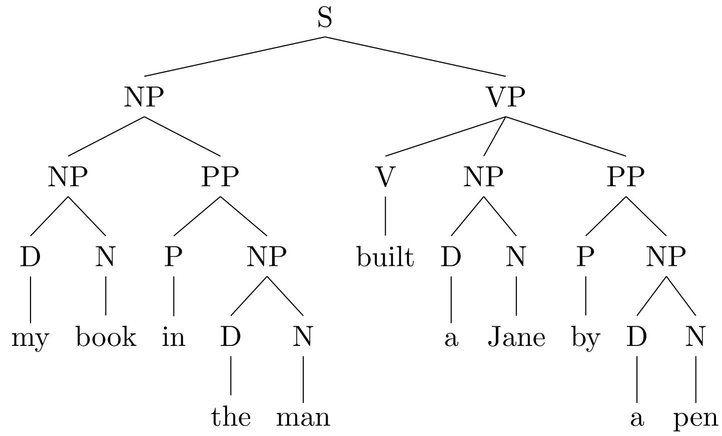
```
S   → NP VP
VP  → V NP | V NP PP
PP  → P NP
NP  → N | D N | NP PP
V   → wrote | built | constructed
D   → a | an | the | my
N   → John | Mary | Jane | man | book | automata | pen | class
P   → in | on | by | with
```

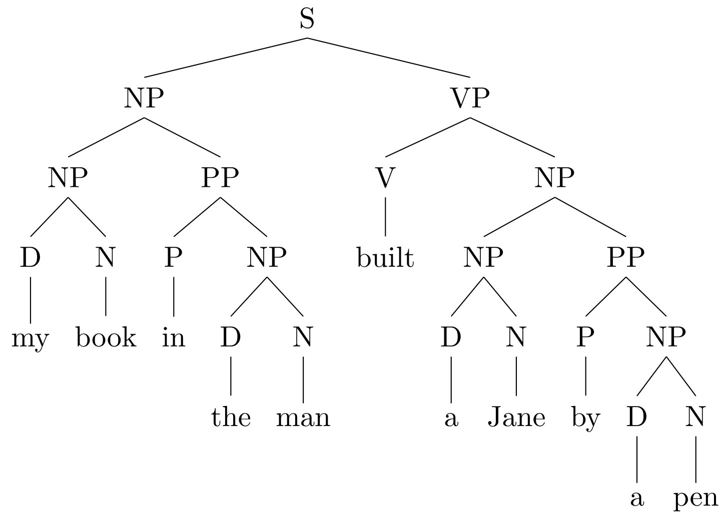
a) Jane constructed automata with a pen (4/20 pts)





b) my book in the man built a Jane by a pen (4/20 pts)





Given the CFG below, answer **c**, **d** and **e**

```
S  → E
E  → E + T | E - T | T
T  → T * I | T / I | I
I  → 0 | 1 | 2 | 3 | 4 | 6 | 7 | 8 | 9
```

c) Provide the left-most derivation of 7 - 4 * 3 step-by-step and plot the final parse tree matching that derivation (4/20 pts)

$$S \xrightarrow{\text{L}} E \xrightarrow{\text{L}} E - T \xrightarrow{\text{L}} T - T \xrightarrow{\text{L}} I - T \xrightarrow{\text{L}} 7 - T \xrightarrow{\text{L}} 7 - T * I \xrightarrow{\text{L}} 7 - I * I \xrightarrow{\text{L}} 7 - 4 * I \xrightarrow{\text{L}} 7 - 4 * 3$$

d) Provide the right-most derivation of 7 - 4 * 3 step-by-step and plot the final parse tree matching that derivation (4/20 pts)

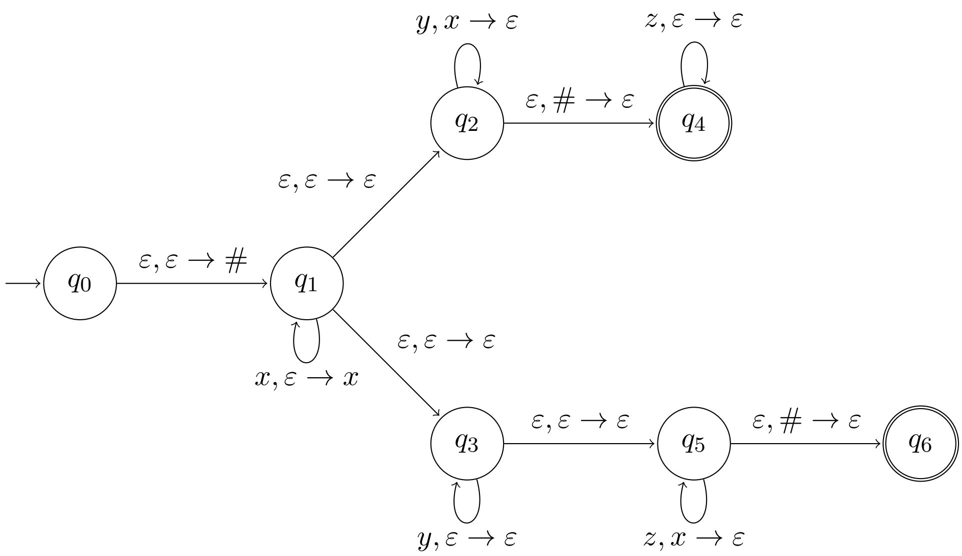
$$S \xrightarrow{\text{R}} E \xrightarrow{\text{R}} E - T \xrightarrow{\text{R}} E - T * I \xrightarrow{\text{R}} E - T * 3 \xrightarrow{\text{R}} E - I * 3 \xrightarrow{\text{R}} E - 4 * 3 \xrightarrow{\text{R}} T - 4 * 3 \xrightarrow{\text{R}} I - 4 * 3$$
$$\xrightarrow{\text{R}} 7 - 4 * 3$$

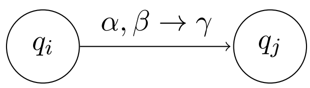
e) Are the derivations in **c** and **d** in the same similarity class? (4/20 pts)

The leftmost derivation and rigthmost derivation of expression 7 - 4 * 3 are in the same similarity class.
Two derivations are similar if one of them can be transformed to other one by a sequence 'switching'.Such a switching can be applied to derivations which are related by precedence relation $\prec$.
Two derivation are related by $\prec$ if they are in the same reflexive,symmetric and transitive closure of $\prec$.
In this expression 7 - 4 * 3,let leftmost derivation is called as D_1 and rigthmost derivation is called as D_n.Since,two derivations are shown with the same parse tree,there exist D_i's where $1 < i < n$ and these D_i's also are in the same similarity class.They make available such 'switchings' D_1 up to D_n.

3 Pushdown Automata (30 pts)

a) Find the language recognized by the PDA given below (5/30 pts)



where the transition $((q_i, \alpha, \beta), (q_j, \gamma))$ is represented as: 

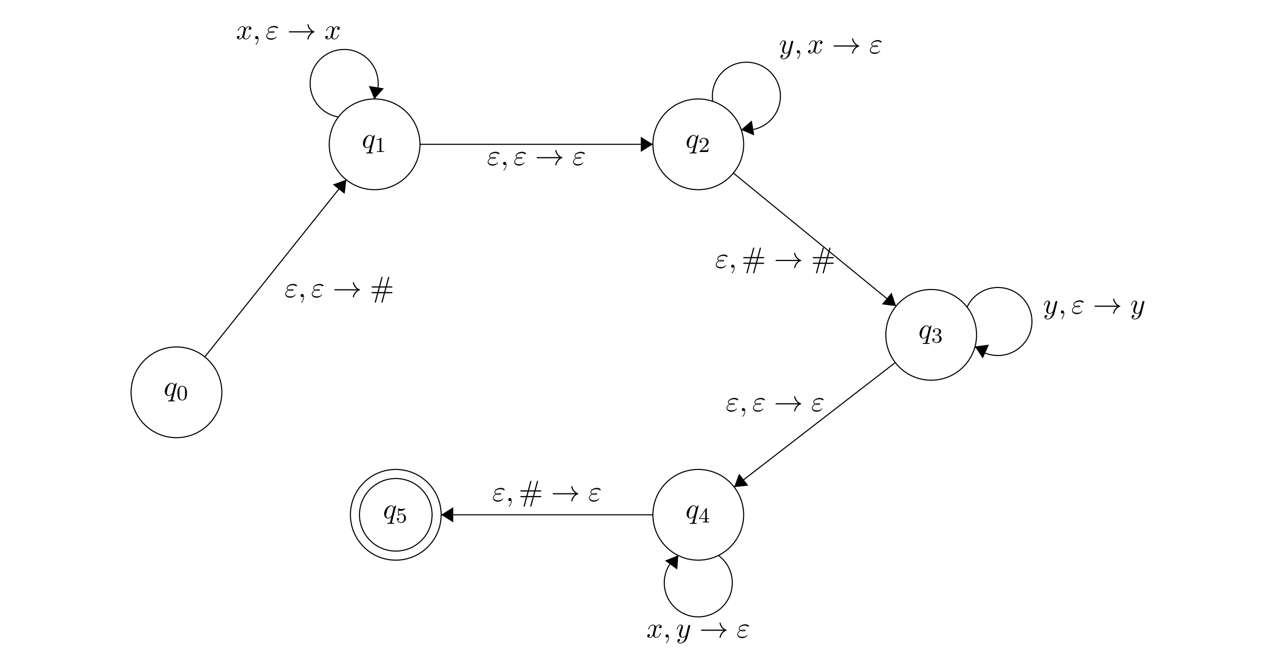
First,this automata pushes a # in stack (and clearly it should be in bottom while other characters are pushing in stack).This # will be used as end of the stack identifier.

After than,in q_1, (x^*) is read.Now, there exist 2 choices.
1-Automata nondeteministically selects q_2 state.
2-Automata nondeteministically selects q_3 state.

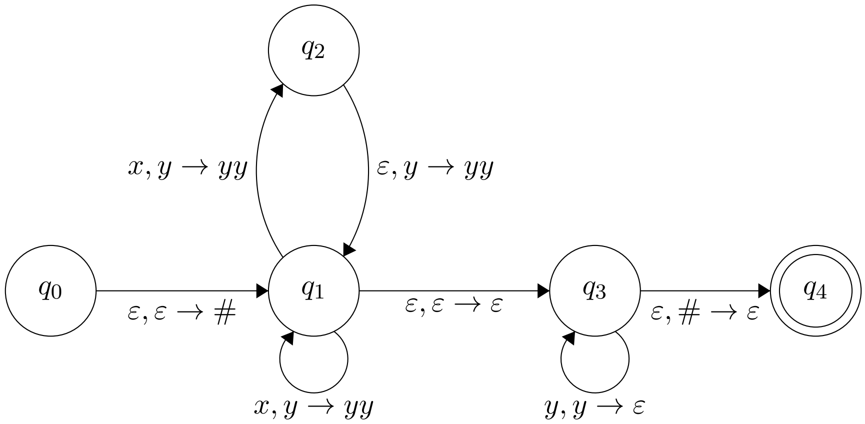
In q_2 state,automata pops an 'x' for each read 'y'.At this step,there is no chance to read 'x','z' or 'y' more than number of read 'x'.When the only character in stack is '#', automata nondeteministically goes to q_4 state by reading ε and pops '#' from stack.In q_4,(z^*) is read. q_4 is a final state so if a string is read up to q_4,it is accepted by automata.
So,let the language accepted by upper part of automata (q_2 is picked) is L_1, then $L_1 = \{x^n y^n z^m : n \geq 0, m \geq 0\}$.
Also automata can pick q_3.
In q_4 state,(y^*) is read and nothing is pushed to the stack.After reading y's, automata goes state q_5.In q_5 state,automata pops an 'x' for each read 'z'.At this step,there is no chance to read 'x','y' or 'z' more than number of read 'x'.When the only character in stack is '#', automata nondeteministically goes to q_6 state by reading ε and pops '#' from stack. q_6 is a final state so if a string is read up to q_6,it is accepted by automata.
So,let the language accepted by lower part of automata (q_3 is picked) is L_2, then $L_2 = \{x^n y^m z^n : n \geq 0, m \geq 0\}$.

Thus,the language recognized by the PDA is $L_1 \cup L_2$ which is $L_1 \cup L_2 = L = \{\{x^n y^n z^m : n \geq 0, m \geq 0\} \cup \{x^n y^m z^n : n \geq 0, m \geq 0\}\}$.

b) Design a PDA to recognize language $L = \{x^n y^{m+n} x^m \mid\ n, m \geq 0;\ n, m \in \mathbb{N}\}$ (5/30 pts)



c) Design a PDA to recognize language $L = \{x^n y^m \mid n < m \leq 2n;\ n, m \in \mathbb{N}^+\}$ (10/30 pts)
Do not use multi-symbol push/pop operations in your transitions.
Simulate the PDA on strings xxy (with only one rejecting derivation) and $xxyyyy$ (accepting derivation) with transition tables.

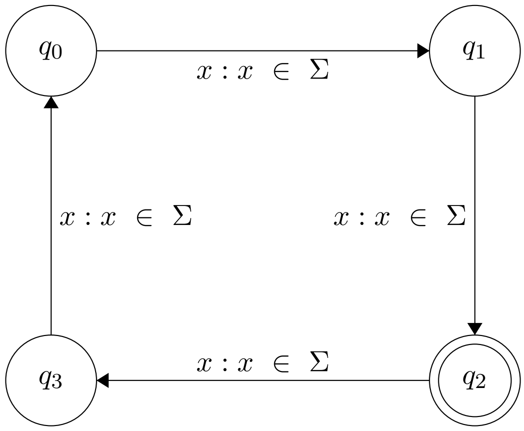


d) Given two languages L' and L as $L' = \{w \mid\ w \in L;\ |w| = 4n + 2\ for\ n \in \mathbb{N}\}$ (10/30 pts)
If L is a CFL, show that L' is also a CFL by constructing an automaton for L' in terms of another automaton that recognizes L.

Like the proof of how intersection of one Context Free Language and a regular language is another Context Free Language,another pushdown automaton can be constructed by taking the steps in the proof.

It is given that L is CFl,so there should be exist a PDA that rexognizes L.Let this PDA is called as M_1,then
$M_1 = \{K_1, \Sigma_1, \Gamma_1, \Delta_1, S_1, F_1\}$ such that
K_1 is the finite set of states,
Σ_1 is input symbols,
Γ_1 is stack alphabet,
Δ_1 is the transition rules,
$S_1 \in K_1$ is the initial state and
$F_1 \subseteq K_1$ is the set of accepting states.

Now,suppose that language L_1 exists and $L_1 = \{|w| = 4n + 2\ \ for\ \ n \in \mathbb{N}\}$.This language is a regular language.Why this language is regular can be shown by creating a DFA.



Let this DFA called as M_2,then
$M_2 = \{K_2, \Sigma_2, \delta_2, S_2, F_2\}$ such that
K_2 is the finite set of states,
Σ_2 is input symbols,
δ_2 is the transition rules,
$S_2 \in K_2$ is the initial state and
$F_2 \subseteq K_2$ is the set of accepting states.

By closure property of intersection of a CFL and a regular language is a CFL as well, L' is a CFL.Since every CFL is read by a PDA,there exist a PDA for L'.
Let the PDA for L' called M, then M;
$M = (K, \Sigma_1, \Gamma, \Delta, S, F)$ where
$K = K_1 \times K_2$
$\Gamma = \Gamma_1$
$S = (S_1, S_2)$
$F = F_1 \times F_2$ and
Δ transition relation $= (((q_1, q_2), \alpha, \beta), (p_1, p_2, \gamma))$ if and only if
both $((q_1, \alpha, \beta), (p_1, \gamma)) \in \Delta_1$ and $((q_2, \alpha), (p_1)) \in \delta_2$.
If pushdown automata has ε transitions,
Δ transition relation $= (((q_1, q_2), \varepsilon, \beta), (p_1, q_2, \gamma))$ if and only if
both $((q_1, \varepsilon, \beta), (p_1, \gamma)) \in \Delta_1$.
Since,in every state,DFA accepts every character from alphabet,transitions in the DFA is not important,transitions in $L' = L(M)$ come from the $L = L(M_1)$,however the important part of the DFA is it's accepting state.A word can be in $L = L(M_1)$ but if this word is not accepted by $L_1 = L(M_2)$,then so does $L' = L(M)$.

So,the costructed PDA M is intersection of both $L = L(M_1)$ and $L_1 = L(M_2)$ and a word 'w' is in $L' = L(M)$ iff $w \in (L(M_1) \cap L(M_2))$.

4 Closure Properties (20 pts)

Let L_1 and L_2 be context-free languages which are not regular, and let L_3 be a regular language. Determine whether the following languages are necessarily CFLs or not. If they need to be context-free, explain your reasoning. If not, give one example where the language is a CFL and a counter example where the language is not a CFL.

a) $L_4 = L_1 \cap (L_2 \setminus L_3)$ (10/20 pts)

L4 is not necessarily be a Context Free Language.

$L2 \backslash L3 = L2 \cap \bar{L}3$(by De Morgan's)
Since regular languages closed under comlementation,$\bar{L}3$ is a regular language as well.

Suppose that $L2 = L(M_1)$ for a pushdown automaton $M_1 = (K_1, \Sigma, \Gamma_1, \Delta_1, S_1, F_1)$ and $\bar{L}3 = L(M_2)$ for a deterministic finite automaton(since every nondeterministic automata has an equal deterministic one,there should be exist a DFA to recognize this language) $M_2 = (K_2, \Sigma, \delta_2, S_2, F_2)$.In order to prove that intersection of L2 and $\bar{L}3$ is context free as well,a new pushdown automata can be constructed.
Let $L2 \cap \bar{L}3 = L(M)$ and pushdown automata $M = (K, \Sigma, \Gamma, \Delta, S, F)$ where
$K = K_1 \times K_2$
$\Gamma = \Gamma_1$
$S = (S_1, S_2)$
$F = F_1 \times F_2$ and
Δ transition relation $= (((q_1, q_2), \alpha, \beta), (p_1, p_2, \gamma))$ if and only if
both $((q_1, \alpha, \beta), (p_1, \gamma)) \in \Delta_1$ and $((q_2, \alpha), (p_1)) \in \delta_2$.
If pushdown automata has ε transitions,
Δ transition relation $= (((q_1, q_2), \varepsilon, \beta), (p_1, q_2, \gamma))$ if and only if
both $((q_1, \varepsilon, \beta), (p_1, \gamma)) \in \Delta_1$.
So,the a word 'w' is in L(M) iff $w \in (L(M_1) \cap L(M_2))$.
Thus, it is proved that $L2 \cap \bar{L}3$ is context free language.Let this language be 'L'.

Now,the question turns into whether $L_1 \cap L$ is context free or not.(Both languages are context free).In order to prove that intersection of 2 Context Free Language is not necessarily be CFL as well,a counter example can be given.
Let $L1 = \{a^n b^n c^m : n, m \geq 0\}$ and $L = \{a^n b^m c^m : n, m \geq 0\}$.(To show that these languages are context free,grammar for both can be given as:
Rules for L1:$S \rightarrow S1S2; S1 \rightarrow aS1b|\varepsilon; S2 \rightarrow cS2|\varepsilon$,
Rules for L:$S \rightarrow S1S2; S1 \rightarrow aS1|\varepsilon; S2 \rightarrow bS2c|\varepsilon$).
$L1 \cap L = \{a^n b^n c^n : n \geq 0\}$.Assume that $L1 \cap L$ is CFL and $w \in L1 \cap L$.(length of w= $|w| > \phi(G)^{|V-\Sigma|}$)
By pumping lemma,this word w can be split into 5 part namely u,v,x,y,z.Then there exist an integer k and $k > (\frac{\phi(G)^{|V-\Sigma|}}{3})$ such that $w = a^k b^k c^k$,clearly,$k < |w|$.
And there exist a split such that:
$w = uvxyz$, $|vxy| \leq k$ and $|vy| \geq 1$.Then,$w = uv^ixy^iz$ should be hold for every $i \geq 0$.
5 possible partitions are:
1-vxy part consist of all a's ;2-vxy part consist of all b's;3-vxy part consist of all c's;4-vxy part

consist of a's and b's;5-vxy part consist of b's and c's.
In first 3,if i is chosen as 0,the word becomes $w = a^{k-l}b^kc^k$ or $w = a^kb^{k-l}c^k$ or $w = a^kb^kc^{k-l}$ respectively and $w \notin L1 \cap L$.(l is the total length of v and y parts)
In last 2 possibilities,if i is chosen as 0,the word w becomes $w = a^{k-l}b^{k-l}c^k$ or $w = a^kb^{k-l}c^{k-l}$ and clearly both generated $w \notin L1 \cap L$.(l is the total length of v and y parts)
Thus,by using pumping lemma,the language $L1 \cap L = \{a^nb^nc^n : n \geq 0\}$ is not a Context Free Language.

In a nutshell,firstly is is proved that $L2 \backslash L3$ is a context free language by using De Morgan Laws and constructing a pushdown automata for it.Afterwards,it is demonstrated that intersection of 2 context free language is not necessarily a Context Free Language.In order to show this,pumping lemma for CFL and counter example are used.

L4 is not necessarily be a Context Free Language.

b) $L_5 = (L_1 \cap L_3)^*$ (10/20 pts)

L5 is necessarily be a Context Free Language.

First $L1 \cap$ Ł3 should be analyzed. Suppose that $L1 = L(M_1)$ for a pushdown automaton $M_1 = (K_1, \Sigma, \Gamma_1, \Delta_1, S_1, F_1)$ and $L3 = L(M_2)$ for a deterministic finite automaton(since every nondeterministic automata has an equal deterministic one,there should be exist a DFA to recognize this language) $M_2 = (K_2, \Sigma, \delta_2, S_2, F_2)$.In order to prove that intersection of $L1$ and $L3$ is context free as well,a new pushdown automata can be constructed.
Let $L1 \cap L3 = L(M)$ and pushdown automata $M = (K, \Sigma, \Gamma, \Delta, S, F)$ where
$K = K_1 \times K_2$
$\Gamma = \Gamma_1$
$S = (S_1, S_2)$
$F = F_1 \times F_2$ and
Δ transition relation $= (((q_1, q_2), \alpha, \beta), (p_1, p_2, \gamma))$ if and only if
both $((q_1, \alpha, \beta), (p_1, \gamma)) \in \Delta_1$ and $((q_2, \alpha), (p_1)) \in \delta_2$.
If pushdown automata has ε transitions,
Δ transition relation $= (((q_1, q_2), \varepsilon, \beta), (p_1, q_2, \gamma))$ if and only if
both $((q_1, \varepsilon, \beta), (p_1, \gamma)) \in \Delta_1$.
So,the a word 'w' is in L(M) iff $w \in (L(M_1) \cap L(M_2))$.
Thus, it is proved that $L1 \cap L3$ is context free language.Let this language be 'L'.

Now,the question becomes whether $L_5 = (L_1 \cap L_3)^* = L^*$ is context free language or not.It is known that L is CFL.By using closure property of CFL under 'kleene star',it can be proved that L^* is CFL.
Let language L is generated by grammar G,namely $L = L(G)$ and $G = \{V, \Sigma, R, S\}$.
A new grammar G_1 for L^* can be written by using grammar G.
$L^* = L(G_1)$ and $G_1 = \{V \cup \{S_1\}, \Sigma, R \cup \{S_1 \to \varepsilon, S_1 \to S_1S\}, S_1\}$. What we have done to obtain G_1 are:

-adding a new starting point to G and adding rule $\{S_1 \to \varepsilon\}$.By this rule,this grammar can create a word $\{\}$ like 'kleene star';
-also add the rule $\{S_1 \to S_1 S\}$ for creating a word as many long as we want by using the rules in grammar G (so does 'kleene star').

To sum up,firstly, it is demonstrated that $L1 \cap$ Ł3 is Context Free Language by constructing a pushdown automata to recognize the language $L1 \cap$ Ł3.Afterwards,by using Context Free Languages are closed under 'kleene star' property,it is proved that $L_5 = (L_1 \cap L_3)^*$ is Context Free Language as well.

Thus, $L_5 = (L_1 \cap L_3)^*$ is necessarily be a CFL.

5 Pumping Theorem (20 pts)

a) Show that $L = \{a^n m^n t^i \mid n \leq i \leq 2n\}$ is not a Context Free Language (10/20 pts) using Pumping Theorem for CFLs.

$L = \{a^n m^n t^i \mid n \leq i \leq 2n\}$ is not a Context Free Language.

Assume that L is a CFL and w is a word $\in L$.(length of w = $|w| > \phi(G)^{|V-\Sigma|}$)
By pumping lemma,this word w can be split into 5 part namely u,v,x,y,z.Then there exist an integer k such that $w = a^k m^k t^i$,clearly,$k < |w|$.
And there exist a split such that:
$w = uvxyz$, $|vxy| \leq k$ and $|vy| \geq 1$.Then,$w = uv^j xy^j z$ should be hold for every $j \geq 0$.
5 possible partitions are:

1-vxy part consist of all a's:

Then,$u = a^l$, $v = a^p$,$x = a^r$, $y = a^s$ and $z = a^h m^k t^i$ such that:
$l + p + r + s + h = k$,$p + r + s \leq k$ and $p + s \geq 1$.
Pumping lemma states that $w = uv^j xy^j z$ should be hold for every $j \geq 0$.
Let j is taken as 0.Then,$u = a^l$, $v = (a^p)^0 = 0$,$x = a^r$, $y = (a^s)^0 = 0$ and $z = a^h m^k t^i$.
The word w becomes $w = a^{k-p-s} m^k t^i$.It is clear that number of a's in this word is not equal to number of m's.So,new $w \notin L$.

2-vxy part consist of all m's;

Then,$u = a^k m^h$, $v = m^p$,$x = m^r$, $y = m^s$ and $z = m^l t^i$ such that:
$l + p + r + s + h = k$,$p + r + s \leq k$ and $p + s \geq 1$.
Pumping lemma states that $w = uv^j xy^j z$ should be hold for every $j \geq 0$.
Let j is taken as 0.Then,$u = a^k m^h$, $v = (m^p)^0 = 0$,$x = m^r$, $y = (m^s)^0 = 0$ and $z = t^i$.
The word w becomes $w = a^k m^{k-p-s} t^i$.It is clear that number of a's in this word is not equal to number of m's.So,new $w \notin L$.

3-vxy part consist of all t's;

Then,$u = a^k m^k t^h$, $v = t^l$,$x = t^p$, $y = t^r$ and $z = t^s$ such that:
$l + p + r + s + h = i$ and $l + r \geq 1$.
Pumping lemma states that $w = uv^j xy^j$ should be hold for every $j \geq 0$.
Let j is taken as $(2 \times n + 1)$.Then,$u = a^k m^k t^h$, $v = (t^l)^{2n+1}$,$x = t^p$, $y = (t^r)^{2n+1}$ and $z = t^s$.
The word w becomes $w = a^k m^k t^{i+(l\times 2n)+(r\times 2n)}$.Since i at least as big as n ($n \leq i$) and at least one of t^l or t^r part is not empty,namely($|l| + |r|$) ≥ 1,the pumped w has at least i+2n t's in it.This is unsuitable because there is a given constraint that i should less than 2n. So,new $w \notin L$.

4-v and y part consist of same number of a's and m's respectively;

Firstly,in $w = uvxyz$, it is clear that if v and y part has unequal number of a and m's,the generated word w after pumping does not have equal number of a's and m's in it.This

is clearly not in language L.
If v and y part consist of same number of a's and m's,then for any $j \geq i$,since v and y parts cannot be empty at the same time,namely($|l| + |r|) \geq 1$,the 'pumped' word definitely have more a's and m's than t's in it.However,this is against to constraint $n \leq i$.Thus,there is no chance to $w \in L$.

5-v part includes m's and y part includes t's;

There are 3 possibilities as well in such a split,which are $|v| = |y|$, $|v| \leq |y|$ and $|v| \geq |y|$.

If $|v| = |y|$,that is if number of m's in v part equal to number of t's in y part,$n \leq i$ constraint is conserved and number of a's and number of m's in w are less than number of t's.Although $n \leq i$ is conserved,then number of a's and number of m's are not equal to each other.
If $|v| \leq |y|$, both number of a's and number of m's are not equal to each other in w and for big j's(pumping number),unfortunately i will pass 2n.
If $|v| \geq |y|$, both number of a's and number of m's are not equal to each other in w and for big j's(pumping number),unfortunately n will pass i.
Thus,there is no chance to $w \in L$.

The only solution which does not harm the constraints ($n \leq i \leq 2n$) and number of a's and m's should be equal in w, is like that:
$w = uvxyz$, v part consist of same number of a and m's and y part has the same length of v part.
However,in such split length of vxy part exceeds the limit.To put a finer point on it,as I mentioned at the top,there exist an integer k which should be greater than $|vxy|$ part.

Thus,there is no split ends up with a word which is in language and does not damage the constraints.At the top,it is assumed that this language L is a CFL but by pumping lemma, it was shown that all possible parititons lead to contradiction.So,L is not a CFL.

b) Show that $L = \{a^n b^{2n} a^n \mid n \in \mathbb{N}+\}$ is not a Context Free Language using Pumping Theorem for CFLs. (10/20 pts)

$L = \{a^n b^{2n} a^n \mid n \in \mathbb{N}+\}$ is not a Context Free Language.

Assume that L is a CFL and w is a word $\in L$.(length of w $= |w| > \phi(G)^{|V-\Sigma|}$)
By pumping lemma,this word w can be split into 5 part namely u,v,x,y,z.Then there exist an integer k and such that $w = a^k b^{2k} a^k$,clearly,$k < |w|$.
And there exist a split such that:
$w = uvxyz$, $|vxy| \leq k$ and $|vy| \geq 1$.Then,$w = uv^jxy^jz$ should be hold for every $j \geq 0$.
5 possible partitions are:

1-vxy part consist of first group a's:

Then,$u = a^l$, $v = a^p$,$x = a^r$, $y = a^s$ and $z = a^h b^{2k} a^k$ such that:
$l + p + r + s + h = k$,$p + r + s \leq k$ and $p + s \geq 1$.
Pumping lemma states that $w = uv^jxy^jz$ should be hold for every $j \geq 0$.
Let j is taken as 0.Then,$u = a^l$, $v = (a^p)^0 = 0$,$x = a^r$, $y = (a^s)^0 = 0$ and $z = a^h b^{2k} a^k$.
The word w becomes $w = a^{k-p-s} b^{2k} a^k$.It is clear that number of a's before b's in this word is not equal to number of a's after b's.So,new $w \notin L$.

2-vxy part consist of b's;

Then,$u = a^k b^l$, $v = b^p$,$x = b^r$, $y = b^s$ and $z = b^h a^k$ such that:
$p + r + s + l + h = 2k$,$p + r + s \leq k$ and $p + s \geq 1$.
Pumping lemma states that $w = uv^jxy^jz$ should be hold for every $j \geq 0$.
Let j is taken as 0.Then,$u = a^k b^l$, $v = (b^p)^0 = 0$,$x = b^r$, $y = (b^s)^0 = 0$ and $z = b^h a^k$.
The word w becomes $w = a^k b^{2k-p-s} a^k$.It is clear that total number of a's in this word is not equal to number of b's.So,new $w \notin L$.

3-vxy part consist of second group a's;

Then,$u = a^k b^{2k} a^h$, $v = a^l$,$x = a^p$, $y = a^r$ and $z = a^s$ such that:
$h + l + p + r + s = i$, $l + p + r \leq k$ and $l + r \geq 1$.
Pumping lemma states that $w = uv^jxy^j$ should be hold for every $j \geq 0$.
Let j is taken as 0.Then,$u = a^k b^{2k} a^h$, $v = (a^l)^0 = 0$,$x = a^p$, $y = (a^r)^0 = 0$ and $z = a^s$.
The word w becomes $w = a^k b^{2k} a^{k-l-r}$.It is clear that number of a's after b's in this word is not equal to number of a's before b's.So,new $w \notin L$.

4-v part includes a's and y part includes b's with length $2 \times |v|$;

Firstly,in $w = uvxyz$, it is clear that if v and y part don't have this fraction,the generated word w after pumping does not conserve number of a's before b's is half of b's in it.This is clearly not in language L.
If v and y part consist of a's and b's respectively with y part's length is 2 times of v,then for any $j \geq i$,since v and y parts cannot be empty at the same time,namely$(|l| + |r|) \geq 1$,the 'pumped' word definitely have more a's before b's than a's after b's in it.However,this is against to constraint that number of a's in two end in the word is same.Thus,there is no

chance to $w \in L$.

5-v part includes b's with length $2 \times |y|$ and y part includes a's ;

Firstly,in $w = uvxyz$, it is clear that if v and y part don't have this fraction,the generated word w after pumping does not conserve number of a's after b's is half of b's in it.This is clearly not in language L.
If v and y part consist of b's and a's respectively with v part's length is 2 times of y,then for any $j \geq i$,since v and y parts cannot be empty at the same time,namely$(|l| + |r|) \geq 1$,the 'pumped' word definitely have more a's after b's than a's before b's in it.However,this is against to constraint that number of a's in two end in the word is same.Thus,there is no chance to $w \in L$.

The only solution which does not harm the constraints total number of b's and a's are equal and number of a's at the two end of word should be equal in w, is like that:
$w = uvxyz$, v part consist of a and b's where number of b's is 2 times of a's and y part has the same property of v part.(begin b's and after a's,number of b's is 2 times of number of a's)
However,in such split length of vxy part exceeds the limit.To put a finer point on it,as I mentioned at the top,there exist an integer k which should be greater than $|vxy|$ part but in this split whole b's in vxy part and also there exist a's in this part.$2k$ cannot be less than vxy part.

Thus,there is no split ends up with a word which is in language and does not damage the constraints.At the top,it is assumed that this language L is a CFL but by pumping lemma, it was shown that all possible parititons lead to contradiction.So,L is not a CFL.

6 CNF and CYK (not graded)

a) Convert the given context-free grammar to Chomsky Normal Form.

$S \to XSX \mid xY$
$X \to Y \mid S$
$Y \to z \mid \varepsilon$

answer here ...

b) Use the grammar below to parse the given sentence using Cocke–Younger–Kasami algorithm. Plot the parse trees.

S → NP VP
S → X1 VP
X1 → Aux NP
S → book | include | prefer
S → Verb NP
S → X2 PP
S → Verb PP
S → VP PP
NP → I | she | me | Houston
NP → Det Nom
Nom → book | flight | meal | money
Nom → Nom Noun
Nom → Nom PP

VP → book | include | prefer
VP → Verb NP
VP → X2 PP
X2 → Verb NP
VP → Verb PP
VP → VP PP
PP → Prep NP
Det → that | this | the | a
Noun → book | flight | meal | money
Verb → book | include | prefer
Aux → does
Prep → from | to | on | near | through

book the flight through Houston

Empty parse table:

1:5 → 1:1 2:5 1:5 → 1:2 3:5 1:5 → 1:3 4:5 1:5 → 1:4 5:5

1:4 → 1:1 2:4 1:4 → 1:2 3:4 1:4 → 1:3 4:4	2:5 → 2:2 3:5 2:5 → 2:3 4:5 2:5 → 2:4 5:5

1:3 → 1:1 2:3 1:3 → 1:2 3:3	2:4 → 2:2 3:4 2:4 → 2:3 4:4	3:5 → 3:3 4:5 3:5 → 3:4 5:5

1:2 → 1:1 2:2	2:3 → 2:2 3:3	3:4 → 3:3 4:4	4:5 → 4:4 5:5

1:1	2:2	3:3	4:4	5:5
book	the	flight	through	Houston

rest of the answer here ...

7 Deterministic Pushdown Automata (not graded)

Provide a DPDA to recognize the given languages, the DPDA must read its entire input and finish with an empty stack.

a) $a^*bc \cup a^n b^n c$

answer here ...

b) $(aa)^* c \cup a^n b^n c$

answer here ...