	_	_
4	,	

Let A and B be two co-turing recognizable languages.

=> There exists turing machines that recognizes the complement of these languages.

Let these be TMMA, TMMB.

L(MA) = A, L(MB) = B.

Assume A ∩B = \$ \$ \$ UB = ≥\*.

Construct a Marhine M is a decider that separates A and B.

Bulid a Turing Machine that will do as follow:

M = 'on input w.

1. Run madines Ma and Ma in parallel.

2. If MB accepts W, output Accept.

3. If Ma accepts w, output reject.

Amording to preliminaries we know that any string in Et is either on A or B.

M uses recognizers for A and B

⇒ It will halt on every japut

A and B are disjoints

⇒ Any string in A 1s in B. (A = B)

Every string in B is accepted by M. ⇒ A ≤ L(m)

If any string in Bis in A. (BEA)

But every string in A 15 rejected by M.

⇒ B is not in the language of M.

Thus, L(M) = c is decider that separates A and B.

## 4.26

Assume PAL ofa = 2(M) Mis a DFA that accepts some palindrome.

If there is a Turing Machine can be presented for the given DFA that runs finitely and halts, then the PALDFA is decidable.

Construct a decider X for PALDFA and a Turny Machine Y that can decides EUFG:

Bulid a Turing Machino that will do as follow:  X = "On import < M>.  In standard a POA P1 such that LIP2 = LIP2 = LIP2   O L(M).  In constant a POA P2 such that LIP2 = LIP2   O L(M).  I convert P2 1 ato an equivalent CFG G.  4. Use Turing Machine Y to observe of LLG2 is empty.  5. If LLG2 is empty then output reject.  6. Else, output assept.  For Twing Machine Y:  Sup 1 and 2 can be done in finite steps. Sup 3 takes finite steps to convert P2 into its equivalent UFG. For large 4, the decider Y abouts if the language LLG2 is empty or LLG2 is not empty. It can be done in a finite step.  So, we give that X tokes finite steps for any input, it's a decider.  Thus, PAL ora is decidable.  5.21  The string ti, tistis ai, aisas; has at least two derivations from T and B. When P has a match with ti, tistis = bi, bisbis.  If the CFG A is ambiguous, then some string s has multiple derivations.  Check the grammar G, it can be seen that derivation of T and B can each generate strings at most one string. Similar, let it be s.  The derivations of S will be as follow.  S = T. S = tjm tjmtj; aj; ajzajm.  S = B, S = bjm bjmbj; aj; ajzajm.  So. We get that tjm tjmtj; = bjm bjmbji, that is a match of P.  All in all, P has a match iff G is ambiguous.  Thus, the reduction from Post Correspondence Problem to AMBIG cra works.  AMBIG cra is undecodable.		
X = "On input CM>.  1. Construct a POA P. South that LIP = twi wis a palendome]  3. Lonstruct a POA P. South that LIP = twi wis a palendome]  3. Lonstruct A POA P. South that LIP = LIP   N L(M).  4. Convert P. Into an equivalent CFG G.  4. Mue Turning Mauline Y to other if LIG   15 empty.  5. If LIG   15 empty then intent reject.  6. Elee, output alcept.  For Turing Mauline Y:  Step 1 and 2 can be done in finite steps. Step 3 takes finite steps to convert P. Into its equivalent CFG. For Turing Mauline Y:  Step 1 and 2 can be done in finite steps. Step 3 takes finite steps to convert P. Into its equivalent CFG. For Step 1 and 2 can be done in a finite step.  So, we get that X takes finite steps for any input, it's a decider.  Thus, PALora is decidable.  5.21  The string ti, tistis. ai. aisai. has at least two derivations from T and B. When P has a match with ti, tistis. bi bisbis.  If the UFG G is ambiguous, then some string s has multiple derivations.  Check the grammar G, it can be seen that derivation of T and B can each generale strings at most one string Smilar, let it be s.  The derivations of S will be as follow.  S ⇒ T . S = tjm tjmtji aji ajzajm  S → B . S = bjm bjmtji aji ajzajm  So, we get that tjm tjmtji = bjbjthat is a match of P.  All in all, P has a match iff a is ambiguous.  Thus, the reduction from Post Correspondence Problem to AMBIG croworks.	Bulid a Turing Machine that will do as follow:	
1. Construct a PDA Pi south that LIPD = LIPD N L(M).  2. Convert P2 1250 an equivalent CFG G.  4. Use Twing Machine Y to other if LiGD is empty.  3. If LiGD is empty then output reject.  6. Elie, output accept.  For Twing Machine Y:  Step 1 and 2 can be done in finite steps. Step3 tokes finite steps to convert P2 into its equivalent CFG. For Ivery at the decider Y wholes if the language LiGD is empty or LiGD is not empty. It can be done in a finite step.  So, we get that X tokes finite steps for any input, it's a decider.  Thus, PAL ora is decidable.  5.21  The string ti, tisti. = bi, bisbi.  If the UFG G is ambiguous, then some string s has multiple derivations from T and B. When P has a match with ti, tisti. = bi, bisbi.  If the UFG G is ambiguous, then some string s has multiple derivations.  Check the grammar G, it can be seen that derivation of T and B can each generate strings at most one string smilar, let it be s.  The derivations of S will be as follow.  S ⇒ T. S = tjm tjm tj. aj. ajajaj  S > B. S = bjm tjmtj. aj. ajaj  So, we get that tjm tjmtj. = bjm tjmbjthat is a match of P.  All in all, P has a match iff a is ambiguous.  Thus, the reduction from Part Correspondence Problem to AMBIG con works.		
4. Use Twong Machine Y to observe if L(G) is empty.  3. If L(G) is empty then output reject.  6. Else, content accept.  For Twing Machine Y:  Step 1 and 2 can be done in finite steps. Step 3 takes finite steps to convert P2 into its equivalent UFG. For (tep 4, the decider Y observe if the language L(G) is compty or L(G) is not compty. It can be done in a finite step.  4. The decider Y observe inite steps for any input, it's a decider.  Thus, PAL ora is decidable.  5.21  The string ti, tisti, ai, aizai, has at least two derivations from T and B. When P has a match with ti, tisti, a in aizbi.  If the UFG G is ambiguous, then some string s has multiple derivations.  Check the grammar G, it can be seen that derivation of T and B can each generate strings at most one string Smilar, let it be S.  The derivations of S will be as follow.  S ⇒ T . S = tjn tjmtj1 ajı ajıajm  S → B . S = bjm bjmbj1 ajı ajıajm  So, we get that tjm tjmtj1 = bjm bjmbj1, that is a match of P.  All in all, P has a match iff a is ambiguous.  Thus, the reduction from Post Correspondence Problem to AMBIG cfu works.		
4. Use Turing Machine Y to other if L(a) is empty.  5. If L(a) is empty then mitpot reject.  6. Else, integrit accept.  For Turing Machine Y:  Step 1 and 2 can be done in finite steps. Step 3 takes finite steps to convert P2 into its equivalent UFG. For Step 4 and 2 can be done in finite steps to L(a) is mot empty. It can be done in a finite step.  So, we got that X takes finite steps for any input, it's a decider.  Thus, PALOFA is decidable.  5.21  The string ti, tisti. aii aizaii has at least two derivations from T and B. When P has a match with ti, tiztibi, bizbis.  If the UFG a is ambiguous, then some string s has multiple derivations.  Check the grammar a, it can be seen that derivation of T and B can each generate strings at most one string smilar, let it be s.  The derivations of S will be as follow.  S \Rightarrow T. S = tim timetil aii aizaim.  So, we get that tim timetil = bim bimebil aii aizaim.  So, we get that tim timetil = bim bimebil aiiaimbilthat is a match of P.  All in all, P has a match iff a is ambiguous.  Thus, the reduction from Post Correspondence Problem to AMBIA crew works.	2. bastrut a POA P2 such that L(B) = L(P1) N L(M).	
4. Use Turing Machine Y to other if L(a) is empty.  5. If L(a) is empty then mitpot reject.  6. Else, integrit accept.  For Turing Machine Y:  Step 1 and 2 can be done in finite steps. Step 3 takes finite steps to convert P2 into its equivalent UFG. For Step 4 and 2 can be done in finite steps to L(a) is mot empty. It can be done in a finite step.  So, we got that X takes finite steps for any input, it's a decider.  Thus, PALOFA is decidable.  5.21  The string ti, tisti. aii aizaii has at least two derivations from T and B. When P has a match with ti, tiztibi, bizbis.  If the UFG a is ambiguous, then some string s has multiple derivations.  Check the grammar a, it can be seen that derivation of T and B can each generate strings at most one string smilar, let it be s.  The derivations of S will be as follow.  S \Rightarrow T. S = tim timetil aii aizaim.  So, we get that tim timetil = bim bimebil aii aizaim.  So, we get that tim timetil = bim bimebil aiiaimbilthat is a match of P.  All in all, P has a match iff a is ambiguous.  Thus, the reduction from Post Correspondence Problem to AMBIA crew works.	3. Convert P2 1ato an equivalent CFG G.	
b. Else, output allept.  For Twing Machine Y:  Step 1 and 2 can be done in finite steps. Step 3 takes finite steps to convert P2 into its equivalent UFG. For Step 4, the decider Y wheels if the language L(G) is compty or L(G) is not compty. It can be done in a finite step.  So, we got that X takes finite steps for any input, it's a decider.  Thus, PAL pen is decidable.  5.21  The string ti, tiztiz ai, aizaic has at least two derivations from T and B. When P has a match with ti, tiztiz = bi, bizbic.  If the cfa G is ambiguous, then some string s has multiple derivations.  Check the grammar G, it can be seen that derivation of T and B can each generate strings at most one string Smillar, let it be s.  The derivations of S will be as follow.  S⇒T. S = tjm tjm tj1 aj1 aj2ajm  S⇒B. S = bjm bjm tj1 aj1 aj2ajm  So, we got that tjm tjm tj1 = bjm bjm bj1. that is a match of P.  All in all, P has a match iff a1s ambiguous.  Thus, the reduction from Post Correspondence Problem to AMBIG cra works.		
For Twing Machine Y:  Step 1 and 2 can be done in finite steps. Step 3 takes finite steps to convert P2 into its equivalent UFG. For (tep 4, the decider Y wheels if the language LIG) is empty or LIG) is not empty. It can be done in a finite step.  Go, we get that X takes finite steps for any input, it's a decider.  Thus, PALOFA is decidable.  5.21  The string ti, tiztie = bi, bizbie  If the string ti, tiztie = bi, bizbie  If the UFG G is ambiguous, then some string s has multiple derivations.  Check the grammar G, it can be seen that derivation of T and B can each generate strings at most one string Smilar, let it be s.  The derivations of S will be as follow:  S > T . S = tjm tjmtj1 aj1 aj2ajm  S > B . S = bjm bjmbj1 aj1 aj2ajm  So, We get that tjm tjmtj1 = bjm bjmbj1. that is a match of P.  All in all, P has a match iff G1s ambiguous.  Thus, the reduction from Post Correspondence Problem to AMBIGGER works.	S. If LCa) is empty then output reject.	
Step 4 and 2 can be done in finite steps. Step 3 takes finite steps to convert P2 into its equivalent UFG. For Keep 4, the decider Y checks if the language L(G) is empty or L(G) is not empty. It can be done in a finite step.  40, we got that X takes finite steps for any input, it's a decider.  Thus, PAL off is decidable.  5.21  The string ti, ti2ti2 = bi, bi2bi.  If the eff G is ambiguous, then some string s has multiple derivations.  Check the grammar G, it can be seen that derivation of T and B can each generate strings at most one string Smilar, let it be s.  The derivations of S will be as follow.  S ⇒ T , S = tjm tjm1tj1.aj1.aj2ajm  S > B , S = bjm bjm1bj1.aj1.aj2ajm  So, we got that tjm tjm1tj1 = bjm bjm1bj1. that is a match of P.  All in all, P has a match iff G1s ambiguous.  Thus, the reduction from Post Correspondence Problem to AMBIG cfa works.	6. Else, antent assept.	
step.  40, we got that X tokes finite steps for any input, it's a decider.  Thus, PALDFA is decidable.  5.21  The string ti, tizti. ai, aizai, has at least two derivations from T and B. When P has a match with ti, tizti = bi, bizbi.  If the UFA a is ambiguous, then some string ≤ has multiple derivations.  Check the grammar G, it can be seen that derivation of T and B can each generale strings at most one string Smilar, let it be ≤.  The derivations of S will be as follow.  S ⇒ T . S = tjm tjm tj, aj, aj 2 aj m  So, we got that tjm tjm tj, aj, aj 2 aj m  So, we got that tjm tjm tj, = bjm bjm bj, . that is a match of P.  All in all, P has a match iff a is ambiguous.  Thus, the reduction from Post Correspondence Problem to AMBIG con works.	For Twing Machine Y:	
Step.  40, we got that X takes finite steps for any input, it's a decider.  Thus, PALOFA is decidable.  5.21  The string ti, tizts, ai, aizai, has at least two derivations from T and B. When P has a match with ti, tiztic = bi, bizbi.  If the cfa G is ambiguous, then some string s has multiple derivations.  Check the grammar G, it can be seen that derivation of T and B can each generate strings at most one string smilar, let it be s.  The derivations of S will be as follow.  S \Rightarrow T, S = tjm tjm-1tj, aj, ajzajm  S \Rightarrow B, S = bjm bjm-1bj, aj, ajzajm  So, we got that tjm tjm-1tj, = bjm bjm-1bj, that is a match of P.  All in all, P has a match iff a is ambiguous.  Thus, the reduction from Post Correspondence Problem to AMBIGGER works.	Step 1 and 2 can be done in finite steps. Step 3 takes finite steps to convert P2 into its equivalent UFG.	For
(so, we got that X takes finite steps for any input, it's a decider.  Thus, Palora is decidable.  5.21  The string ti, tizti, ai, aizai, has at least two derivations from T and B. When P has a match with ti, tizti = bi, bizbi.  If the cfa a is ambiguous, then some string s has multiple derivations.  Check the grammar a, it can be seen that derivation of T and B can each generate strings at most one string smilar, let it be s.  The derivations of S will be as follow.  S ⇒ T , S = tjm tjm-1 tj, aj, ajz ajm  S ⇒ B , S = bjm bjm-1 bj, aj, ajz ajm  So, we got that tjm tjm-1 tj, = bjm bjm-1 bj, that is a match of P.  All in all, P has a match iff a is ambiguous.  Thus, the reduction from Post Correspondence Problem to AMBIG cfa works.	Stup 4, the decider Y checks if the language L(G) is empty or L(G) is not empty. It can be done in a	inite
Thus, PALOFA is decodable.  5.21  The string ti, tiztr. ai, aizai, has at least two derivations from T and B. When P has a match with ti, tiztiz = bi, bizbiz  If the CFG a is ambiguous, then some string s has multiple derivations.  Check the grammar a, it can be seen that derivation of T and B can each generate strings at most one string smilar, let it be s.  The derivations of s will be as follow.  S \Rightarrow T. S = tjm tjm-1 \cdots tj, aj, ajz ajm  S \Rightarrow B. S = bjm bjm-1 \cdots bj, aj, ajz ajm  So, we got that tjm tjm-1 \cdots tj, a ambiguous.  Thus, the reduction from Post Correspondence Problem to AMBIG cfg works.	Step.	
The string ti, tizti. ai, aizai. has at least two derivations from T and B. When P has a match with ti, tizti. = bi, bizbi.  If the UF a a is ambiguous, then some string s has multiple derivations.  Check the grammar a, it can be seen that derivation of T and B can each generate strings at most one string smilar, let it be s.  The derivations of s will be as follow.  S \Rightarrow T. S = tjm tjm1 \cdots tj, ajzajm  S \Rightarrow B. S = bjm bjm1 \cdots bj, ajzajm  So, we get that tjm tjm1 \cdots tj, ajzajm  So, we get that tjm tjm1 \cdots tj, ajzajm  So, we get that tjm tjm1 \cdots tj, ajzajm  So, we get that tjm tjm1 \cdots tj, ambiguous.  Thus, the reduction from Post Correspondence Problem to AMBIG cfa works.	Go, we got that X tookes finite steps for any input, it's a decider.	
The string ti, tizti, ai, aizai, has at least two derivations from T and B. When P has a match with ti, tizti, = bi, bizbi.  If the cfa G is ambiguous, then some string s has multiple derivations.  Check the grammar G, it can be seen that derivation of T and B can each generate strings at most one string similar, let it be s.  The derivations of S will be as follow.  S \Rightarrow T, S = tjm tjm-1tj, aj, aj,ajm  S \Rightarrow B, S = bjm bjm-1bj, aj, aj,ajm  So, we get that tjm tjm-1tj, = bjm bjm-1bj, . that is a match of P.  All in all, P has a match iff a is ambiguous.  Thus, the reduction from Post Correspondence Problem to AMBIG cfa works.	Thus, PALDFA is devidable.	
The string ti, tizti, ai, aizai, has at least two derivations from T and B. When P has a match with ti, tizti, = bi, bizbi.  If the cfa G is ambiguous, then some string s has multiple derivations.  Check the grammar G, it can be seen that derivation of T and B can each generate strings at most one string similar, let it be s.  The derivations of S will be as follow.  S \Rightarrow T, S = tjm tjm-1tj, aj, aj,ajm  S \Rightarrow B, S = bjm bjm-1bj, aj, aj,ajm  So, we get that tjm tjm-1tj, = bjm bjm-1bj, . that is a match of P.  All in all, P has a match iff a is ambiguous.  Thus, the reduction from Post Correspondence Problem to AMBIG cfa works.		
match with ti, tistil = bi, bizbil  If the CFG G is ambiguous, then some string s has multiple derivations.  Check the grammar G, it can be seen that derivation of T and B can each generate strings at most one string similar, let it be s.  The derivations of S will be as follow:  S >> T . S = tjm tjm-1 tj. aj. aj ajm  S >> B . S = bjm bjm-1 bj. aj. aj ajm  So, we get that tjm tjm-1 tj. = bjm bjm-1 bj that is a match of P.  All in all, P has a match iff G is ambiguous.  Thus, the reduction from Post Correspondence Problem to AMBIG cfg works.		
If the UFG G is ambiguous, then some string s has multiple derivations.  Check the grammar G, it can be seen that derivation of T and B can each generate strings at most one string smilar, let it be s.  The derivations of S will be as follow:  S > T. S = tjm tjm-1 ··· tj1 aj1 aj2 ··· ajm  S > B . S = bjm bjm-1 ··· bj1 aj1 aj2 ··· ajm  So, we get that tjm tjm-1 ··· tj1 = bjm bjm-1 ··· bj1. that is a match of P.  All in all, P has a match iff a1s ambiguous.  Thus, the reduction from Post Correspondence Problem to AMBIG cf4 works.	•	4
Check the grammar $G$ , it can be seen that derivation of $T$ and $B$ can each generate strings at most one string smilar, let it be $S$ .  The derivations of $S$ will be as follow: $S \Rightarrow T$ , $S = t_1^m t_1^m \cdots t_1^m t_1^m t_1^m t_2^m t_2^m t_3^m t$		
one string Similar, let it be \$.  The derivations of \$\( \) will be as follow:  \$\( S \Rightarrow T   S = t_{\sqrt{m}} t_{\sqrt{m}} \cdots \cdot t_{\sqrt{1}} a_{\sqrt{1}} a_{\sqrt{2}} \cdots a_{\sqrt{m}} \\ \$\( S \Rightarrow T   S = b_{\sqrt{m}} b_{\sqrt{m}} \cdots \cdots b_{\sqrt{1}} a_{\sqrt{2}} \cdots a_{\sqrt{m}} \\ \$\( S \rightarrow B   S = b_{\sqrt{m}} b_{\sqrt{m}} \cdots \cdots a_{\sqrt{m}} a_{\sqrt{m}} a_{\sqrt{m}} \cdots a_{\sqrt{m}} a_{\s		
The derivations of S will be as follow: $S \Rightarrow T$ , $S = t_j^m t_j^m \cdots t_j^n \Delta_j^n \Delta_j^n \cdots \Delta_j^m$ $S \Rightarrow B$ , $S = b_j^m b_j^m - \cdots b_j^n \Delta_j^n \Delta_j^n \cdots \Delta_j^m$ $S \Rightarrow B$ , $S = b_j^m b_j^m - \cdots b_j^n \Delta_j^n \Delta_j^n \cdots \Delta_j^n \cdots \Delta_j^n \Delta_j^n \cdots \Delta_j^n \Delta_j^n \cdots \Delta_j^n \Delta_j^n \cdots \Delta_j^n \Delta_j^n \Delta_j^n \cdots \Delta_j^n \Delta_j$		most
S⇒T. S=tjm tjm-1···tj1 aj1 aj2···ajm S⇒B. S=bjm bjm-1···bj1 aj1 aj2···ajm So, we got that tjm tjm-1···tj1 = bjm bjm-1···bj1. that is a match of P. All in all, P has a match iff a1s ambiguous. Thus, the reduction from Post Correspondence Problem to AMBIGGEN works.		
S ⇒ B . S = bjm bjm-1 ··· bj1 Aj1 Aj2 ··· Ajm So, we got that tjm tjm-1 ··· tj1 = bjm bjm-1 ··· bj1. that is a match of P. All in all, P has a match iff G1s ambiguous. Thus, the reduction from Post Correspondence Problem to AMBIG cf4 works.	· · · · · · · · · · · · · · · · · · ·	
So, we got that tim tim 1 ··· til = bim bim-1 ··· bil . that is a match of P.  All in all, P has a match iff G is ambiguous.  Thus, the reduction from Post Correspondence Problem to AMBIG cra works.		
All in all, P has a match iff Gis ambiguous.  Thus, the reduction from Post Correspondence Problem to AMBIGGG works.		
Thus, the reduction from Post Correspondence Problem to AMBIGGG works.		
AMBIGUFA is undecidable.		
	AMBILICFA is undecidable.	

Let A be the language [ <m,x>  Mis a Tm and M doesn't accept x].</m,x>
By reduction from ATM, it can be checked that A is not Turing-recognizable.
First, we will reduce A to J.
Ning reduction function fly) = 14
So, we got that y is in A if and only if fly) is in J.
As we know, the fuction f is computable, and A is not Turing-recognizable.
Thus, Jis not Twing-recognizable.
After that, we will reduce ATM to J.
Nsing reduction function 9 Ly) = Dy.
So, we got that y is in ATM if and only if gly) is in J
Also, as we know that the function q is computable and Arm is not Turing - recognizable.
Thus, Jis not Turing - recognizable.
5.25
If there exist a Turing mauhine that accepts and halts on every input string of the language
then that language is known as Decidable or Recursive. All the decidable language is also Turing—
Aueptable. If the language of all yes instances to A is decidable then decision problem A is also
decidable. Any Twing-recognizable works instead of co-Turing-recognizable language works (or vice versa).

5.30 (c) ALLTM = 
$$\frac{1}{2}$$
 M is a TM and L(M) =  $\frac{1}{2}$  Let ALLTM be the language such that ALLTM is a description of Twing Machines.  
It's non-trivial as some TM accepts  $\frac{1}{2}$ , and other do not have it.

It's satisfies the two conditions of rice's theorem.

Some TM's accept all possible string of an alphabet.

For Example: Atm. (xt Atm < m y & ATm)

5.24

If some language is recognized by 2 Turing Machines, either both have descriptions in Allton or neither will have.

Go. Using The theorem prove. Allom 13 undecidable.

6.13
<u> </u>
Given for each m > 1. Zm=20,1,2,, m-1] is a universe. Fm = (zm,+,x) be the model whose universe is Zm. +, x are relations over modulo m.
So, we know that Zm is finite.
Let Qi are quantifius. Φ has no quanti ens.  A formula φ = QiXi Q2X2···· QkXk Φ (Xi,X2····Xn)
•
Define II for 0 < 1 < n
$I_{n}(x_{1}x_{2}\cdots x_{n}) = \varphi(x_{1}x_{2}\cdots x_{n})$
If Qlis H. Il-1 (X1 X2 ··· X1-1)= V 11-0 Ii (X1 X2··· X1-1, 1)
If Quis V. IL-1 (X1X2 XL-1) = 1 (X1X2 XL-1, 1)
⇒ We can simply annmerate all the possible values into formula and the formula is time.
By induction argument it is proved. The theory Th(Fm) is decidable.