Nome: Alok Bhawankor ROIL: PAOG Panel: 1 Assignment 4.

IMPLEMENTATION OF UNIFILATION ALGORITHM AI Aim: To Implement Unification algorithm. Objective: To study and implement Unification algorithm. Theory ? 1) UNIFICATION ALGORITHM · Unification is a programme of making two different logical abonic expressions identical by finding a substitution Unification depends on substitution · Unification algorithm in a recursive (matching) * procedure that compares two literals and discovered whether their excist a set of substitutions that make them identical. . Fach likeral supresented as list, where first element is name of predecate and remaining elements are arguements, which may be single element (ahm). . It return a unifice for the two literals of enists

or returns fail it expression do not match with

each other. This substitution variables are called

Most Grenord Unifer or MGU.

Slep 4: Set Substitution Set (SUBST) to MIL Slep 5: For i= 1 to the number of elements in 31 a) (all Unity function with the 1th elements of 5, and ith elements of 52 and put result in b) IF Sub = failure, then returns FAILURE. C) Tr Jub # MTL, Hendo. Platform: aid why resolution is required? ans: Resolution is used if these are various statements given and we need to prove a conclusion of those statements. Resolution provides a way to look for ground dayses staring from 7 or in clausal form (of - FOL) Clever way to look for proof, as a way to focus 027 What can the pre-requisits for applying Unification algorithm. Ans: Condition for Unification 1) Predicable symbol must be same, alons or expression with different pridicale symbol can never be unified 2) Number of Arguements in both expression must

I'dentical.

Spand To the south them add it to the set of clause available to the procedure. Topul: Topo likeal I and Is Output: A set of substitution Adjustition: Unification Hyposition Step 1: If 3, and 32 is Variable as constant them: a) If 3, and 32 are indentical then between MIT b) Else if 3, is variable a. Then if 31 occurs in 32, settlem FATTURE b. clsc settlem & (5,15,13) c) Else if 32 is variable a If 32 accurs in 34, return FATTURE b else return & (5,15,2)3 d) Else return FATTURE Step 2: If initial Predicate symbol in 3, and 32 are not some return FATTURE		
Resolve them bogether The besulting clouse. Resolve them bogether The besulting clouse. Called Assolvent will be a disjunction of all of the likerals of both of the parent clouses in Society in empty cloure, them construction is found. The it is not them add it has the set of clouse civaliable to the paraceluse. Thout of clouse civaliable to the paraceluse. Thought of Set of substitution. Playsithm: Unification Adjorithm. Styp 1: The sty and So is Variable as constant them: a) The sty and So is variable. a) The sty and So are individual than between the stype if So accurs in So, publish from the constant them. b) Else if So is variable. a) The sty so accurs in So, publish from the constant them. b) Clour them of So accurs in So, publish from the constant them. C) Else if So is variable. a) The stype of several in So, publish from the constant them. Styp 2: The initial Predicate Symbol in So, and So are not some subturn fatture. Skp 3: So and So bruse different numbers of argument than return fatture.	(g) D	how reduction grouph
called Assolven with be a disjunction of all of the likebols of both of the parate Clauses in the School in empty clause. Then contradiction is found. To it is not them add it to the set of clause available to the procedure. Toput: Too likebol I, and Is Output: It set of substitution Plansithm: Unification Algorithm Shap I: It s, and so is Vaniable or constant them: a) It s, and so are indicated then between MIT to Else if s, is variable a. Then if s occurs in so, return from the class if so is variable b. clse seturn & (so s) } C) Else if so is variable a If so accurs in so, return fattures b. clse seturn & (so s) } d) Else seturn FATTURE Shap 3: I initial Predicate symbol in s, and so are not same return fatture. Shap 3: So and so true different number of argument than return fatture.	And the street of the street o	- Select topo clauses Call II
of the likeway of both of the parent clauses in 3 TE Esolvent in empty clause, then contradiction— is found. If it is not them add it to the set of clause available to the procedure— Topul . Too likewal I, and Is Output: A set of substitution. Algorithm: Unification Algorithm Step 1: If 3, and 32 is Noniable as constant them: a) If 5, and 52 is Noniable as constant them: b) Else if 3, is variable a. There if 3 occurs in 52, return PATURE b. Else if 52 is variable a If 52 genera in 54, return FATURE b else return EGS, IS, IS d) Else between FATURE Step 2: If initial Predicate Symbol in 5, and 32 are not some return FATURE.		
3 TF Esolvent in empty clause, then Contradiction is found. The it is not them add it to the set of clause analiable to the princedure. Toput: Two likeal I and Is Output: A set of substitution Algorithm: Unification Algorithm Step 1: The 34 and 32 is Variable on constant them: a) The 34 and 32 is Variable on constant them: b) Else if 34 is Variable a. Then if 31 occurs in 324 network from the disc setting & Sep 1 is substituted b. clse setting & E(3, 13,) 3 C) Else if 32 is variable a The 32 accurs in 34, return FATIVAGE b else return & E(3, 13, 2) 3 d) Else return FATIVAGE Step 2: The initial Predicate symbol in 34 and 32 are not some return fativage Step 3: Si and 32 true different number of argumants than return FATIVAGE		The contract of the contract o
Set of clause available to the procedure Toput: Too likeal I, and Iz Output: A set of substitution Algorithm: Unification Algorithm Ship I: If Stand So is Variable or constant than: a) It stand So are indentical than return MIT b) Else if St is variable a. Then if St occurs in So there is An algorithm c) Else if So is variable a. If So occurs in St, return PATILIRE b. else return E(St I St) 3 d) Else return FATILIRE Ship 2: If initial Predical Symbol in St, and So are not same return FATILIRE	7	THEORY OF WATER
Input: Two likeal I, and Is Output: A set of substitution. Algorithm: Unification Algorithm Styp 1: If 3, and 32 is Variable as constant there a) If 3, and 32 are indentical then between MIT b) Else if 3, is variable a. Then if 3, accurs in 32, return fattures b. clse settem & (52 5) 3 c) Else if 52 is variable a. If 52 accurs in 34, return fattures b. else return & (51 52) 3 d) Else return FATTURE Styp 2: If initial Predicate symbol in 3, and 32 are not same return fatture. Styp 3: 51 and 32 true clifferent number of arguments then return FATTURE		is found. If it is not the contradiction -
Output: A set of substitution. Algunithm: Unification Algorithm Step 1: TF 31 and 52 is Variable ar constant there a) TF 31 and 32 are indentical then between MIT b) Fisc if 31 is variable a. Then if 31 occurs in 32, return FATILIRE b. clsc between £(52 31)3 c) Fisc if 32 is variable a TF 32 occurs in 34, return FATILIRE b else return £(31 32) 3 d) [ise return FATILIRE Step 2: TF initial Predical symbol in 31 and 32 are not some return FATILIRE. Step 3: SI and 32 hour clifferent number of arguments than return FATILIRE.		set of dause avaliable to the procedure.
Algorithm: Unification Algorithm Ship 1: IF 5, and 52 is Variable or constant thin: a) IF 5, and 52 are indentical thin return MIT b) Else if 5, is variable a. Then if 51 occurs in 52, return FATTHERS b. else return & (52 5,1) } c) Else if 52 is variable a IF 52 occurs in 51, return FATTHERS b else return & (5,1 52) } d) Else return FATTHERS Ship 2: If initial Predical symbol in 5, and 52 are not same return FATTHERS Ship 3: 51 and 52 true different number of argumate then return FATTHERS		input: Two likeral Li and La
Step 1: IF 3, and 32 is Variable or constant then: a) The 3, and 32 are indentical then return MIT b) Fise if 3, is variable a. Then if 31 occurs in 32, return PATILIRE b. clae between E(52 51) 3 c) Fise if 32 is variable a. If 32 occurs in 31, return PATILIRE b. else return E(5, 152) 3 d) Fise return FATILIRE Step 2: If initial Predicate 34mbol in 5, and 32 are not some return FATILIRE Step 3: 51 and 52 base different number of arguments then return FATILIRE	0	Julput: 17 set of substitution.
Step 1: IF 3, and 32 is Variable ar constant then: a) IF 3, and 32 are indentical then return MIT b) Fise if 3, is variable a. Then if 31 occurs in 32, return PATILIRE b. clsc return & (52 5,1) } c) Fise if 32 is variable a IF 32 occurs in 34, return PATILIRE b. else return & (5,1 52 3) d) Fise return FATILIRE Step 2: If initial Predical Symbol in 5, and 32 are not same return FATILIRE Step 3: 51 and 52 true different number of arguments then return FATILIRE	.[2]	equalithm: Unification Algorithm
a) It 3, and 32 are indentical then between MIT b) Else if 3, is variable a. Then if 31 occurs in 52, return friture b. clse return & (52 51) 3 c) Else if 32 is variable a. If 32 occurs in 31, return friture b. else return & (51 52) 3 d) Else return FAILURE Step 2: If initial Predicate symbol in 5, and 52 are not some return failure Step 3: 51 and 32 bruse different number of arguments then return failure.	7	Step 1: IF 5, and 52 is Variable as constant these
b) Else if 3, is variable a. Then if 3, occurs in 32, return FATTURE b. clsc return & (5, 5, 3 c) Else if 52 is variable a If 52 occurs in 31, return FATTURE b. else return & (5, 5, 3) d) Else return FATTURE Step 2: If initial Predicate symbol in 3, and 52 are not some return FATTURE Step 3: 51 and 52 have different number of arguments then return FATTURE.		a) IF 3, and 32 are understical then return MIT
b. ilse sekurn & (5, 15,) 3 C) Else if 52 is variable a IF 52 occurs in 31, return FATILIRE b. else sekurn & (5, 15, 2) 3 d) Else sekurn FATILIRE Slep 2: If initial Predicak symbol in 3, and 32 are not some seturn FATILIRE Slep 3: 51 and 52 brink different number of arguments then return FATILIRE		b) Else if 3, is variable
C) Else if 32 is variable a Tf 32 accurs in 31, return FATIMES b else return EATIMES d) Else return FATIMES Step 2: If initial Predicate symbol in 31 and 32 are not some return FATIMES Step 3: SI and 32 true different number of arguments then return FATIMES		a. Then it si occurs in so return FATILIRE
Step 3: Si and 32 how different number of arguments then return FATILIRE		
b else return EAILURE Step 2: If initial Predicate symbol in 5, and 52 are not same return FAILURE Step 3: Si and 32 true different number of arguments then return FAILURE		
d) [tse return FAILURE Step 2: If initial Predicate symbol in 5, and 52 are not some return FAILURE Step 3: 51 and 32 true different number of arguments then return FAILURE		•
Step 3: Si and 32 hove different number of arguments Then return FATIURE.		
Step 3: Si and 32 hove different number of arguments Then return FATIURE.	3	ilep 2 . If initial Predicak symbol in 3, and 32 are
then return FATIURE		
	51	