

29/12/20

AI LAB TEST 2
WRITE UP

M..

AKSHAY MITUR
IBM18CS010

Q Given: $P \Rightarrow Q$ $R \Rightarrow S$

Prove: $P \vee R \Rightarrow Q \vee S$ * (check end for conversion to (NF))

Program for resolution:

~~\Leftarrow kb = []~~

~~def CLEAR():~~

~~global kb~~

~~kb = []~~

~~def TELL(sent):~~

~~global kb~~

~~if isClause(sent): kb.append~~

def disjunctify(clauses):

disjuncts = []

for clause in clauses:

disjuncts.append(tuple(clause.split('v')))

return disjuncts

def getResolvents(c_i, c_j, d_i, d_j):

resolvent = list(c_i) + list(c_j)

resolvent.remove(d_i)

resolvent.remove(d_j)

return tuple(resolvent)

(PTO)

29/12/20

AI LAB WRITEUP TEST 2
IDM18CS010
AKSHAY MITTUR

```
def resolve(Ci, Cj):  
    for di in Ci:  
        for dj in Cj:  
            if di == '~' + dj + dj == '~' + di:  
                return getResolvent(Ci, Cj, di, dj)  
    return [Ci, Cj]
```

```
def checkResolution(clauses, query):  
    clauses += [query if query.startswith('~') else '~' + query]  
    proposition = '^'.join(['(' + clause + ')'] for clause in clauses)  
    clauses = disjunctify(clauses)  
    resolved = False  
    new = set()  
    while not resolved:  
        n = len(clauses)  
        pairs = [(clauses[i], clauses[j]) for i in range(n) for j in range(i+1, n)]  
        for (Ci, Cj) in pairs:  
            resolvent = resolve(Ci, Cj)  
            if not resolvent:  
                resolved = True  
                break  
        new = new.union(set(resolvent))  
        if new.issubset(set(clauses)): break  
    for clause in new:  
        if clause not in clauses:  
            clauses.append(clause)
```

29/12/20

AILAB WRITEUP

TEST 2

IBMIBCSO10

AKSHAY MITTAL

if resolved: print("KB entails query")
else: print("KB doesn't entail query")

Given rules = $\{ P \Rightarrow Q, R \Rightarrow S \}$

T.P. $\Rightarrow P \vee R \Rightarrow Q \vee S$

As given in the question, converting to CNF

$P \vee R \Rightarrow Q \vee S$

$\sim (P \vee R) \vee (Q \vee S)$

~~$\sim P \wedge \sim R$~~ $\sim P \wedge \sim R \vee Q \vee S$

\Leftarrow Query