```prolog
:- style_check(-singleton).
stream('C:/Users/Sandu/Desktop/Project KRR/a.txt').
write_to_file(Stream, Message):-
                        telling(OldStream),tell(Stream),
                        write(Message),
                        told,tell(OldStream).

read_from_file(Stream, X):-
                        seeing(OldStream),see(Stream),
                        read(X),
                        read(end_of_file),
                        seen,see(OldStream).

wtf(Message):-
            telling(OldStream),tell('C:/Users/Sandu/Desktop/Project KRR/b.txt'),
            write(Message),
            told,tell(OldStream).

rff(X):-
            seeing(OldStream),see('C:/Users/Sandu/Desktop/Project KRR/a.txt'),
            read(X),
            read(end_of_file),
            seen,see(OldStream).

resolve(C1,C2,P,Res) :-
        member(P, C1),
        negate(P,Q),
        member(Q, C2),
        delete(C1,P,D1),
        delete(C2,Q,D2),
        subtract(D1,D2,D),
        merge(D,D2,Res),
        !.

resolve(C1,C2,P,Res) :- member(P, C2),
                                    negate(P,Q),
                                    member(Q, C1),
                                    delete(C1,Q,D1),
                                    delete(C2,P,D2),
                                    subtract(D1,D2,D),
                                    merge(D,D2,Res),
                                %dif(Res,[]),
                                    !.


add_to_formula(Res,X1,X):-
                                    \+member(Res,X1),
                                    merge([Res],X1,X).

negate_all([],R,R).
negate_all([H|T],R,R1):-negate(H,P),negate_all(T,[P|R],R1).

get_terms(X,L):-append(X,R),negate_all(R,[],R1),merge(R1,R,R2),sort(R2,L).

search_formula(_,[],_,_,_).
search_formula(P,[C1|T],C1,X,C2):- member(P,C1),search_formula2(n(P),X,C2).
search_formula(P,[H|T],C1,X,C2):- search_formula(P,T,C1,X,C2).

search_formula2(_,[],_):-false.
search_formula2(P,[C1|T],C1):- member(P,C1),!.
search_formula2(P,[H|T],C1):- search_formula2(P,T,C1).
```

```prolog
negate(n(A), A) :- !.
negate(A, n(A)).

rec([[]|_]):-!,wtf('UNSATISFIABLE'),nl,!,false.
rec(X):-
    (search_formula(H,X,C1,X,C2),
    %get_terms(X,L),
    %member(H,L),
    member(C1,X),
    member(C2,X),
    resolve(C1,C2,H,Res),
      add_to_formula(Res,X,F),!;wtf('SATISFIABLE'),nl,!,false),
      rec(F).

solve(X):- rff(X),rec(X).
```