ITCS 6150/8150 Project

OPTION 1: Implement the verification method (called RS-method), described below, of whether a given formula in the propositional calculus is a tautology.

Assume that L_0 is a language of order zero (propositional calculus). Letter S (with indices) will denote finite sequences (α1, α2…., αm) of formulas in L_0.

If S1 = (α1, α2…., αm) and S2 = (β1, β2, …, βn) and α, β are any formulas then S1, α, S2 and S1, α, β, S2 denote sequences (α1, α2…., αm, α, β1, β2, …, βn) and (α1, α2…., αm, α, β, β1, β2, …, βn).

A **formula** in L_0 **is indecomposable** if it is either a propositional variable or negation of propositional variable.

A **sequence is indecomposable** provided it is formed only of indecomposable formulas

A **sequence is fundamental** if it simultaneously contains a formula α and its negation ¬α.

We consider two types of schemas: S1/S2 and S1/(S2; S3). S1 is called is a premise and S2, S3 conclusions. If a schema is of the form S1/(S2;S3), then S2 is left conclusion and S3 right conclusion. The following 7 schemas are considered:

[S1,(α∪β),S2]/[S1,α,β,S2]   [S1,(α∩β),S2]/[S1,α,S2; S1,β,S2]

[S1,¬(α∩β),S2]/[S1,¬α, ¬β ,S2]   [S1,¬(α∪β),S2]/[S1,¬α,S2; S1,¬β,S2]

[S1,(α→β),S2]/[S1,¬α,β,S2]   [S1,¬(α→β),S2]/[S1,α,S2; S1,¬β,S2]

[S1,¬¬α,S2]/ [S1,α,S2], where S1 is indecomposable in all schemas.

Let's denote by D(α) the diagram (tree) built for α using the above seven schemas.

FACT: Formula α is a propositional tautology if and only if all end sequences in the diagram D(α) (leaves in D(α)) are fundamental.

System Input: propositional formula; **System output**: yes/no

----

OPTION 2: Write a program which solves N queens puzzle (8 < N < 13).

Input: N; **Output**: N×N Boolean array with 1's showing final position of queens.

Your program should use heuristics minimizing the search space.