

Travail Pratique – Implémentation de la Résolution Logique par Réfutation

Problème

De nombreux problèmes combinatoires ou de vérification peuvent être modélisés à l'aide de formules en logique propositionnelle. Leur résolution repose sur des méthodes automatiques de démonstration, en particulier la résolution par réfutation, introduite par J. A. Robinson en 1965.

Cette technique consiste à démontrer l'inconsistance d'un ensemble de clauses en dérivant la clause vide, signe de contradiction.

Objectif

Implémenter en langage C un solveur logique fondé sur la résolution par réfutation, permettant de déterminer si un ensemble de clauses (en forme normale conjonctive simplifiée) est satisfiable ou non satisfiable.

Travail demandé

1. Lecture d'un fichier texte contenant l'ensemble de clauses au format FNC simplifié.
2. Implémenter l'algorithme de résolution par réfutation.
3. Tests : créer des tests unitaires pour vérifier la correction du solveur.
4. Implémenter un module supplémentaire permettant d'obtenir la FNC d'une formule
5. Application Pratique : appliquer le solveur à un problème concret.

Livrables

- Rapport : détailler les choix d'implémentation, les optimisations réalisées, et les résultats obtenus. Inclure des captures d'écran, des fichiers de sortie, et des traces de résolution pour illustrer le fonctionnement du solveur.

Practical Work – Logical Resolution solver

Problem

Many combinatorial or verification problems can be modeled using propositional logic formulas. Their resolution relies on automatic proof methods, particularly resolution by refutation, introduced by J. A. Robinson in 1965. This technique involves demonstrating the inconsistency of a set of clauses by deriving the empty clause, indicating a contradiction.

Objective

Implement a logical solver in the C programming language based on resolution by refutation. This solver should determine whether a set of clauses, expressed in simplified conjunctive normal form (CNF), is satisfiable or unsatisfiable.

Required Work

1. **Reading Clauses:** Implement the reading of a text file containing the set of clauses in simplified CNF format.
2. **Algorithm Implementation:**
 - Develop the resolution by refutation algorithm.
3. **Testing:**
 - Create unit tests to verify the correctness of the solver.
4. **Practical Application:**
 - Apply the solver to a concrete problem to demonstrate its utility.
 - Implement an additional module that allows obtaining the CNF of a formula.

Deliverables

- **Report:** detail the implementation choices, optimizations made, and results obtained. Include screenshots, output files, and resolution traces to illustrate the solver's functionality.