Alexander J. Summers

Program Verification

Exercise Sheet 2: Encoding Problems to SAT

Assignment 1 (Eliminating Equality)

The variation of Ackermannization shown in the lectures to eliminate equality potentially generates many variables and extra conjuncts to express instances of the reflexivity, symmetry and transitivity properties of equality.

1. How many variables are introduced, given n term variables in the original formula?
2. How many instances of reflexivity, symmetry and transitivity properties will be added?
3. Can you think of ways to reduce the necessary additions to the original formula?

Assignment 2 (Eliminating Uninterpreted Functions)

Apply the techniques explained in the lectures, to convert the following formula into a formula without the functions f and g (you don't have to convert it all the way to a propositional one, unless you want to):

$$f(g(x)) = x \wedge f(y) = x \wedge \neg(y = g(x))$$

The resulting formula should be satisfiable - what is a model for it? What model does this describe for the *original* formula above?

Assignment 3 (Encoding the Eight Queens Puzzle)

The eight queens puzzle is a classical problem of placing eight queens on a 8×8 chessboard so that no two queens threaten each other[1]. Consider a variation of the eight queens puzzle with 8 rooks[2] instead of 8 queens. Think how you would encode this puzzle as a SAT problem (following the encoding recipe described in the lectures) and answer the following questions:

1. How many variables does your encoding use?

[1] https://en.wikipedia.org/wiki/Eight_queens_puzzle
[2] https://en.wikipedia.org/wiki/Rook_(chess)

2. How many clauses and of what size does your encoding generate?
3. How easy would it be to extend your encoding to support the original eight queens problem?