

Guide to Computing Answer Sets by Hand

Please note that this guide is not meant as a substitute for learning the definition of answer set, nor is it meant to be used without supporting context. All of this information is stated more precisely in the book. This is simply a reminder of some points that can make your life easier.

1. If there are no facts, try the empty set first.
2. If there are non-disjunctive facts, any answer set of the program must contain them all.
3. For disjunctive facts, any answer set must contain at least one of the disjuncts of each fact.
4. Constraints get rid of candidate answer sets which violate them. They do not “remove” literals from sets.
5. Any set S must be minimal with respect to the subset relation if it is to be an answer set. This means that for S to be an answer set, there cannot be another set that is an answer set of the program that is a proper subset of S .
6. Every element of an answer set of a program must be supported by some rule in the program. This means that you can eliminate from consideration any set that contains elements that are not in the head of any rule of the program. Remember that we are guided by the principle that we do not believe in things which we have no reason to believe.
7. When you are looking for an answer set of a program with default negation, you will need to compute the reduct. A reduct is a new program (without default negation) created from the original (which has default negation) by following the rules in Part II of the definition of answer set. You need a candidate set to compute a reduct because a reduct is always found with respect to some set. When searching for an answer set, you may have to compute multiple reducts as you may have to test multiple candidate sets to see if they are answer sets.