

CNF format

A satisfiability problem in conjunctive normal form consists of a the conjunction of a number of *clauses*, where is clause is a disjunction of a number of variables or their negations. If we let x_i represent variables that can assume only the values *true* or *false*, then a sample formula in conjunctive normal form would be

$$(x_1 \vee x_3 \vee \neg x_4) \wedge (x_4) \wedge (x_2 \vee \neg x_3)$$

where \vee represents the *OR* boolean connective, \wedge represents *AND*, and $\neg x_i$ is the negation of x_i .

Given a set of clauses C_1, C_2, \dots, C_m on the variables x_1, x_2, \dots, x_n , the satisfiability problem is to determine if the formula

$$C_1 \wedge C_2 \wedge \dots \wedge C_m$$

is satisfiable. That is, is there an assignment of values to the variables so that the above formula evaluates to *true*? Clearly, this requires each C_j to evaluate to *true*.

To represent an instance of this problem, we will create an input file that contains all of the information needed to define a satisfiability problem. This file will be an ASCII file consisting of a two major sections: the preamble and the clauses.

The Preamble. The preamble contains information about the instance. This information is contained in lines. Each line begins with a single character (followed by a space) that determines the type of line. These types are as follows:

- **Comments.** Comment lines give human-readable information about the file and are ignored by programs. Comment lines appear at the beginning of the preamble. Each comment line begins with a lower-case character **c**.

`c This is an example of a comment line.`

- **Problem line.** There is one problem line per input file. It has the following format.

`p FORMAT VARIABLES CLAUSES`

The lower-case character **p** signifies that this is the problem line. The **FORMAT** field allows programs to determine the format that will be expected, and should be the word “cnf”. The **VARIABLES** field contains an integer value specifying n , the number of variables in the instance. The **CLAUSES** field contains an integer value specifying m , the number of clauses in the instance. This line must occur as the last line of the preamble. Fields are separated by an arbitrary amount of whitespace.

The Clauses. The clauses appear immediately after the problem line. The variables are assumed to be numbered from 1 up to n . It is not necessary that every variable appear in an instance. Each clause will be represented by a sequence of numbers, each separated by at least one space, tab or newline character. The non-negated version of a variable i is represented by i ; the negated version is represented by $-i$.

Each clause is terminated by the value 0. Unlike many formats that represent the end of a clause by a new-line character, this format allows clauses to be on multiple lines.

The last clause may or may not be terminated by the value 0. If it is, any file content following this 0 must be ignored.

Example. Using the example

$$(x_1 \vee x_3 \vee \neg x_4) \wedge (x_4) \wedge (x_2 \vee \neg x_3)$$

Two possible input files would be, for example:

```
c Example CNF format file
c
p cnf 4 3
1 3 -4 0
4 0 2
-3
```

```
p cnf 4 3
1
3
-4 0 4 0 2 -3 0
%
This was another example CNF format file.
Anything after the last clause is ignored.
```