Introduction:

In this class project I have decided to implement a weighted MAX-SAT solver in Prolog. The Maximum Satisfiability problem (MAX-SAT) is the problem of determining the maximum number of clauses, of a given Boolean formula, that can be satisfied by some assignment. The weighted maximum satisfiability problem generalizes MAX-SAT. It asks for the maximum weight, which can be satisfied by any assignment, given a set of weighted clauses. Another common variant in Partial MAX-SAT. The partial maximum satisfiability problem (P-MAX-SAT) asks for the maximum number of clauses, which can be satisfied by any assignment of a given subset of clauses (known as soft clauses). The rest of the clauses must be satisfied (hard clauses). Weighted MAX-SAT generalizes P-MAX-SAT as well. These problems are NP-Complete. I have used CLPR and CLPFD to solve the problem exactly.

Problem Formulation:

The problem is modeled as a Mixed Integer Program (MIP). Given a Max-SAT instance with \( n \) variables and \( r \) clauses, it is translated into a Mixed ILP as follows. We use \( r \) extra continuous variables \( y_j \), one per clause. Each clause \( C_j \) can be encoded as the linear constraint. The function to minimize is the weighted sum

\[
min \sum_{j=1}^{r} w_j y_j
\]

As example, the set of clauses \( \{ \overline{v}_1, \overline{v}_2, v_1 \lor v_2 \} \) generates the following ILP,

\[
min \ (y_1 + y_2 + y_3) \\
1 - v_1 + y_1 \geq 1 \\
1 - v_2 + y_2 \geq 1 \\
v_1 + v_2 + y_3 \geq 1
\]

After the MAX-SAT problem is converted to an MIP, I have used a branch-and-bound based MIP solver present in CLPR. This gives me the exact sum, that we are planning to minimize. This sum is fed to a CLPFD program, which labels the variables that satisfies this cost.

Challenges Faced:

The most difficult challenge in this project was to integrate CLPFD and CLPR. The objective function minimized by CLPR is in Reals, where as CLPFD expects everything as integers. The next hurdle was to read the MAX-SAT input in Dimacs format(http://maxsat.ia.udl.cat/restrictions/) from a file. I have written a small Java program to convert the dimacs file format to a prolog friendly file format.