

Logické řešiče

- CNF a DNF známe z logiky

DPLL

Dostanu klauzule, list proměnných, model = {}

1. Pokud je každá klauzule *true*, vrátím *true*
2. Pokud je každá klauzule *false*, vrátím *false*
3. Pokud existuje Pure-Symbol, nastavím na *true* a zavolám DPLL
 - Pure-Symbol je takový, který je všude se stejným znaménkem
4. Pokud existuje jednotková klauzule nastavím na *true* a zavolám DPLL
 - Jednotková klauzule je taková, která v sobě má jenom jeden literál
5. Vezmu první nesplněný symbol a nastavím na *true* zkusím DPLL
6. Pokud se nepodařilo nastavím na *false* a zkusím DPLL
7. Vrátím *false*

Dopředné a zpětné řetězení

DEF Hornerova klauzule je taková klauzule, že obsahuje nejvýše jeden pozitivní literál tzn $a \wedge (a \vee \neg b) \wedge (b \vee \neg a \vee \neg c)$

Můžeme přepsat na implikace $a \wedge (b \Rightarrow a) \wedge (c \wedge a \Rightarrow b)$.

Nastavíme DAG s proměnnými. Každá hrana odpovídá jedné implikaci. Proměnné si drží hodnotu kolik jejich předpokladů musí být ještě splněno. Pokud je 0, proměnnou nastavíme jako splněnou. Pokaždé, když splníme proměnnou snížíme počítadlo všem proměnným co mě měli jako předpoklad. Nastavíme data na splněno (forward chaining)

Obráceně nastavíme cíl na splněno a následně splníme všechny jeho předpoklady a pak předpoklady předpokladů atd. . .