# Teorema 1 – demonstratie

Pentru a demonstra ca Perfect vs Chance este intractable (nu are un algoritm polinomial eficient care sa resolve toate cazurile), putem sa reducem problema la o alta problema NP-completa. Un exemplu de astfel de problema este 3SAT.

A. Pentru ca 3SAT sa fie NP-completa, este necesar sa fie simultan atat NP, cat si NP-hard.
   1. Daca inlocuim variabilele cu valori booleene, putem verifica in timp linear rezultatul expresiei. Din acest motiv, problema 3SAT este in NP.
   2. Putem arata ca 3SAT este o problema NP-hard daca am putea reduce problema SAT la o problema 3SAT. Acest lucru este posibil pentru orice problema n-SAT prin adaugarea unor variabile auxiliare. Ex:

      Initial in SAT: $(x1 \lor x2 \lor \neg x3 \lor x4 \lor \neg x5 \lor x6 \lor x7)$

      Reducere la 3SAT: $(x1 \lor x2 \lor y1) \land (\neg y1 \lor x4 \lor y2) \land (\neg y2 \lor x6 \lor x7)$

      Astfel, putem afirma ca 3SAT este NP-hard.

   **Din (1) si (2) rezulta faptul ca 3SAT este o problema NP-completa.**

   ( https://www.cs.utep.edu/vladik/cs5315.20/31.pdf)

B. Putem reduce problema Perfect vs Chance la problema 3SAT prin cautarea unei configuratii a variabilelor astfel incat expresia sa fie adevarata, considerand ca fiecare variabila reprezinta o decizie din cadrul problemei Perfect vs Chance. Daca fiecare clauza (grupare de 3 variabile) este adevarata, solutia este Perfect, altfel este Chance.
   1. Reducand problema Perfect vs Chance la 3SAT, putem afirma ca Perfect vs Chance este NP-hard.
   2. De asemenea, verificarea solutiei poate fi realizata in timp polinomial, astfel algoritmul este incadrat si in NP.
   3. Presupunem ca P != NP.

   **Din (1), (2) si (3) rezulta faptul ca problema Perfect vs Chance este intractable.**

   (https://algs4.cs.princeton.edu/66intractability/)