

PROJET MATHÉMATIQUES-INFORMATIQUE : ERGO

DELAY Emmanuel

I. Présentation du jeu

Le point de départ est le jeu **Ergo**, « The Game of Proving You Exist! ». Vous pouvez retrouver les règles détaillées [ici](#).

Le jeu est composé de 55 cartes : 4 de chaque variable (A, B, C ou D), 4 de chaque opérateur (ET, OU, \implies), 6 cartes NON, 8 parenthèses, 3 cartes Ergo et 10 cartes particulières.

Chaque joueur (4 maximum) se voit assigné une variable (A, B, C ou D) au début du jeu. À chaque manche, les joueurs essaient collectivement de créer une preuve de leur existence tout en réfutant l'existence des autres joueurs. À chaque tour, un joueur pioche deux cartes et doit jouer deux cartes (éventuellement les défausser). Lorsque une carte Ergo est jouée ou qu'il n'y a plus de carte dans la pioche la preuve est terminée. À condition qu'il n'y ait pas de paradoxe, chaque joueur dont l'existence est prouvée reçoit un nombre de points égal au nombre de cartes dans la preuve. Toutes les cartes sont ensuite mélangées et une nouvelle manche est lancée. Le premier joueur ayant 50 points gagne.

Concernant la construction de la preuve, un certain nombre de règles doivent être respectées :

- la preuve doit avoir au maximum 4 lignes. Dès qu'elle atteint 4 lignes, toutes les cartes supplémentaires doivent être jouées sur une de ces lignes;
- chaque ligne doit être syntaxiquement correcte (deux opérateurs ou deux variables ne peuvent pas se suivre, chaque parenthèse ouvrante doit correspondre à une parenthèse fermante, ...);
- une carte peut être insérée entre deux cartes déjà posées à condition que le résultat reste syntaxiquement correct.

II. Le projet

Le but est de réaliser une implémentation en Python de ce jeu. Pour cela, je vois plusieurs points à traiter, plus ou moins par ordre de difficulté croissante :

- analyser une ligne de la preuve pour vérifier qu'elle est syntaxiquement correcte;
- coder une ligne syntaxiquement correcte sous une forme exploitable (arbre, forme conjonctive normale, forme disjonctive normale, ...?);
- déterminer à partir du codage des 4 lignes quelles variables sont prouvées ou s'il y a une contradiction;
- réaliser une interface graphique (à priori avec tkinter);
- implémenter une fonction pour pouvoir jouer contre l'ordinateur.

Si on finit tout ça et qu'on a peur de s'ennuyer, on pourra toujours creuser pour améliorer la façon dont l'ordinateur joue. Au pire, on demandera à Frédéric Muller de nous prêter ses TetrisBot pour qu'ils apprennent à jouer à Ergo;-)