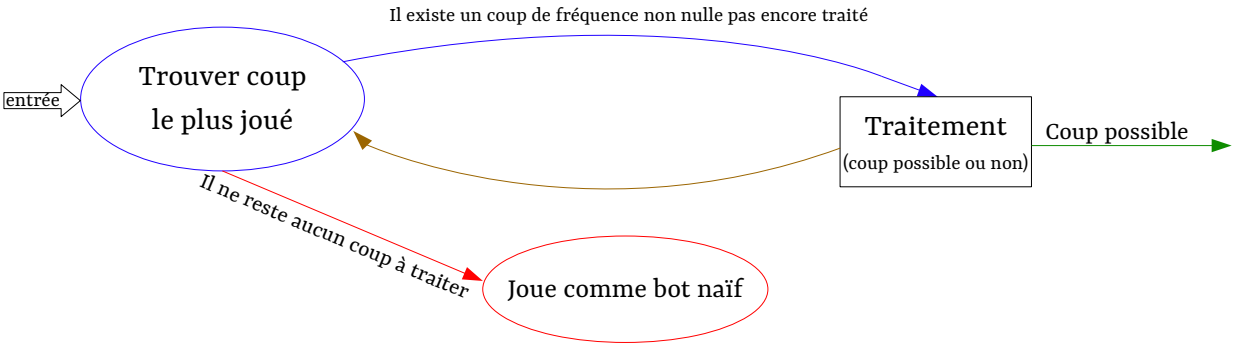
Description de projet personnel : jeu de plateau, bot en apprentissage

I – Premier bot : apprentissage par analyse de fréquence des coups

Le bot a pour objectif de jouer, pour le tour n, le coup le plus joué contre lui au tour n. Il retiendra néanmoins seulement les coups de joueurs qui ont gagné la partie contre lui. Les coups joués pendant la phase de construction seront ceux du bot naïf. Pour stocker les informations, on utilise une liste, contenant des listes de 8 éléments constitués d'un coup (sous la forme de chaîne de caractère) et d'un entier représentant la fréquence associée à ce coup (ces deux données dans une troisième liste à deux éléments). Python permet d'augmenter la taille de la première liste (nombre de lignes de la matrice) au besoin.



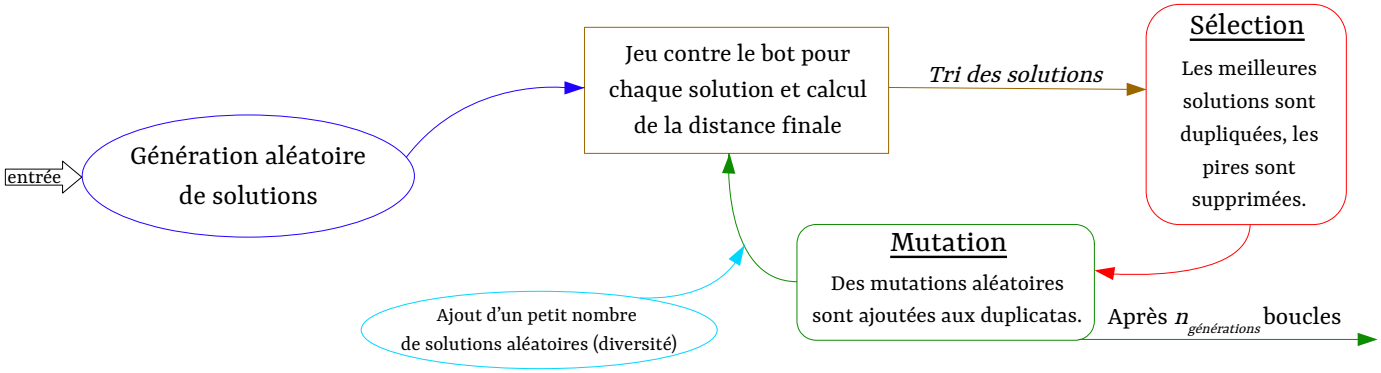
Algorigramme du bot « qui apprend »

II – Second bot : apprentissage par algorithme de sélection génétique

L'établissement d'une solution optimale est limité par le nombre de possibilités élevé pour un coup (rester sur place pour grandement augmenter le *momentum*, détruire un mur, déplacement horizontal ou vertical, déplacements diagonaux). Il y a donc 10 possibilités pour chacun des coups. D'où 10^N suites de N coups possibles. Afin de dépasser cette limite, on peut introduire une notion de distance à l'objectif. Cette distance d : (x,y) → int se définit par induction (dans le cas d'un départ en bas à droite):

- Pour toutes cases C de coordonnées (x,y) de valeur V, d(x,y) = (min(d(x-1,y),d(x,y-1),d(x+1,y),d(x,y+1)) + V)
- d(0,0) = 0

Où (x,y) est la case ligne x, colonne y. La valeur d'une case vaut 0 si elle est vide, ∞ si c'est un joueur (∞ entier strictement supérieur à tout entier) ou un entier naturel correspond à la taille du mur (nombre de « destructions » nécessaires sur la case pour la rendre vide) sinon. On peut alors mettre en place un algorithme qui sélectionne les solutions optimales en fonction du nombre de tours nécessaires pour atteindre le camp de l'adversaire, ou à défaut la distance d après N tours. Une solution est une liste de N coups, N choisit arbitrairement comme un compromis entre temps de calcul et variété des solutions possibles.



Algorigramme de l'algorithme de sélection