

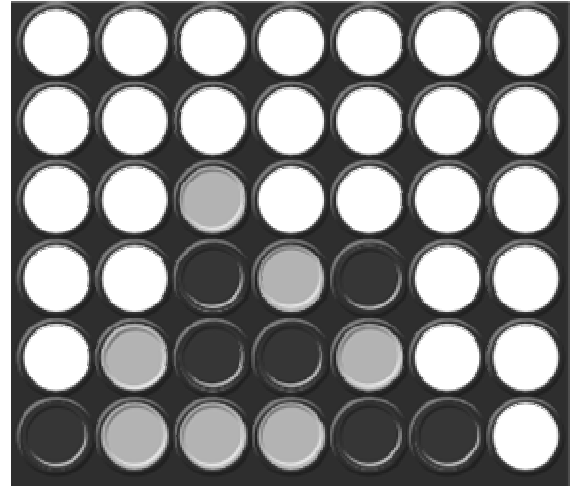


TP PLC : Puissance 4 -jeu-



Wikipedia donne les informations suivantes sur le jeu Puissance 4 (disponible également en chinois) : « Puissance 4 est un jeu de société combinatoire abstrait au tour par tour, édité pour la première fois en 1974 par MB (détenu de nos jours par Hasbro) et qui se joue à 2. Le but est de faire une ligne de 4 pions sur une grille comptant 6 rangées et 7 colonnes, les parties durent en général une dizaine de minutes. L. Victor Allis a étudié ce jeu dans sa thèse intitulée *A Knowledge-based Approach of Connect-Four* (1989). On sait maintenant que la personne qui commence a toujours la possibilité de gagner. [...] »

Règles : Le plateau de jeu est une grille verticale. Lorsque l'on glisse un pion dans une colonne, le pion descend jusqu'en bas. Chaque joueur dispose de 21 pions d'une couleur (jaune ou rouge). Tour à tour les joueurs posent un pion dans une colonne. Celui qui gagne est celui qui aligne le premier quatre pions de sa couleur. »



L'objectif du TP est de réaliser un programme permettant de jouer au jeu Puissance 4 seul contre l'ordinateur.

- Etape 1 (**Modélisation**) : définir une structure de donnée permettant de représenter une grille. Par exemple, vous pourrez utiliser une liste de 7 listes de 6 éléments. Chaque élément pourrait avoir trois valeurs 'noir', 'gris' et 'vide (blanc)', ou être une variable qui s'instanciera seulement avec deux valeurs possibles 'gris' et 'noir'. Dans le premier cas, la grille ci-dessus serait alors définie par : `[[noir, vide, vide, vide, vide, vide], [gris, gris, vide, vide, vide, vide], [gris, noir, noir, gris, vide, vide], [gris, noir, gris, vide, vide, vide], [noir, gris, noir, vide, vide, vide], [noir, vide, vide, vide, vide, vide], [vide, vide, vide, vide, vide, vide]]`. Vous pouvez également considérer une modélisation plus générale sous la forme d'une liste de listes ou une structure en arbre. On peut alors avoir une représentation de la grille ci-dessus plus courte : `[[noir], [gris, gris], [gris, noir, noir, gris], [gris, noir, gris], [noir, gris, noir], [noir]]`. Dans la plupart des cas, il sera préférable de traiter cette structure de donnée en faisant abstraction de sa taille fixe, et généraliser comme si c'était une structure de taille quelconque.
- Etape 2 (**Fin du jeu**) : définir et réaliser un premier prédicat qui détermine si une configuration de la grille correspond à une victoire pour l'un ou l'autre des joueurs.
- Etape 3 (**Coup gagnant**) : définir et réaliser un second prédicat qui détermine si parmi les coups possibles, l'un donne une grille gagnante pour l'un ou l'autre des joueurs.
- Etape 4 (**Coup défensif**) : définir et réaliser un troisième prédicat qui détermine si parmi les coups possibles, l'un donne une grille pour son adversaire avec un coup gagnant.
- Etape 5 : **Généralisation** des étapes 3 et 4.
- Etape 6 : **Conclure**.

Remarque 1 : si nécessaire vous pouvez simplifier les règles du jeu (pour l'étape 1 et 2).

Remarque 2 : si vous le voulez vous pouvez prendre un autre jeu (un jeu similaire pour appliquer la même démarche, la démarche présentée ici s'appliquant dans bon nombre de jeux).

Remarque 3 : l'utilisation des notions abordées dans la seconde partie de ce semestre (contraintes, gel, coupure, ...) sera appréciée, en particulier lorsqu'elle sera justifiée explicitement.