

Bases de la Programmation

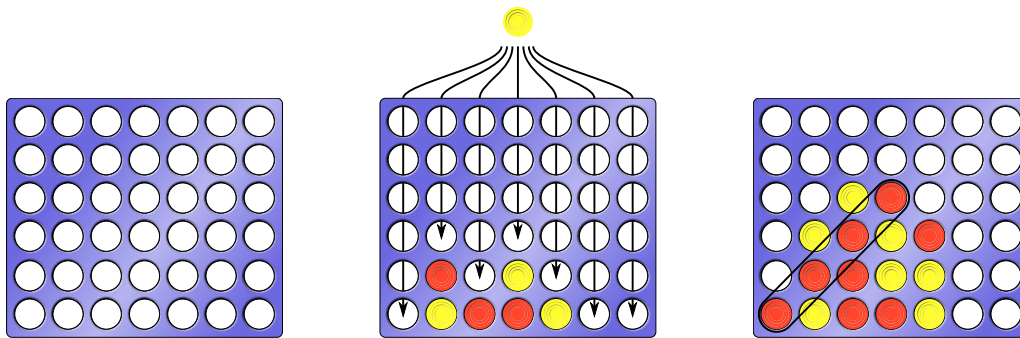
TP 9 – Puissance 4

V. Padovani, PPS - IRIF

Le jeu de Puissance 4 se joue à deux joueurs, Rouge et Jaune, sur un plateau vertical de 6×7 cases (Figure gauche ci-dessous). Chaque joueur dispose de 21 pions de sa couleur.

Les joueurs jouent à tour de rôle. Le joueur dont c'est le tour choisit une colonne qui n'est pas encore pleine, et y insère un de ses pions. Le pion tombe jusqu'à ce qu'il se retrouve bloqué par un autre pion ou par le bas de la colonne (Figure centrale).

La partie se termine dès que l'un des deux joueurs parvient à aligner quatre pions de sa couleur horizontalement, verticalement ou en diagonale (Figure droite) – le joueur est alors déclaré gagnant – ou lorsque le plateau est plein – la partie est alors déclarée nulle.



La modélisation de ce jeu en C ne nécessite qu'une simple matrice d'éléments d'un type énuméré à trois valeurs (vide, pion rouge, pion jaune). Le choix des dimensions du plateau est facilement modulable à l'aide de constantes (`H` et `W` ci-dessous),

La fonction centrale du jeu est évidemment très facile à écrire : la colonne choisie par un joueur ne doit pas être pleine, et la position de chute d'un pion est calculable par une simple boucle. Un peu plus délicate est l'écriture de la vérification de l'existence d'un gagnant :

```
typedef enum {
    DRAW,           // pas de gagnant
    R_WINS,         // rouge gagne
    Y_WINS,         // jaune gagne
} result;

result winner (content b[H][W]);
```

Trouvez le moyen d'écrire cette seule fonction en minimisant les redondances dans votre code, de manière concise, correcte et efficace. Ce n'est pas totalement immédiat. Il vous faudra sans doute un ou plusieurs nouveaux types, ainsi que des fonctions auxiliaires.