

# Synthèse : Puissance 4 (en F)

## 1 puissance4

Le jeu « Puissance 4 » est proposé par MB Jeux. Sur la boîte, on peut lire les indications suivantes.

Puissance 4 est un jeu de stratégie verticale passionnant et plein d’astuce. C’est un jeu facile à apprendre et amusant à jouer. Ses règles sont simples. Chaque joueur essaie de faire une rangée de quatre pions dans le cadre – horizontalement, verticalement ou diagonalement – tout en essayant d’empêcher son adversaire de faire de même. Croyez-moi ce n’est pas aussi facile que cela en a l’air ! La position verticale du jeu demande au joueur beaucoup de concentration et de réflexion.

### Montage :

1. Suivez les instructions illustrées pour l’assemblage des parties en plastiques (non reproduites ici).
2. Placez le jeu entre les joueurs.
3. Chaque joueur choisit une couleur de pions.

### But du jeu :

Être le premier joueur à placer quatre de ses pions sur une ligne horizontale, verticale ou diagonale continue.

### Règles du jeu :

1. Choisissez le premier joueur. Le joueur qui commence la première partie sera le deuxième joueur au cours de la deuxième partie.
2. À tour de rôle, chaque joueur fait tomber un de ses pions dans une des fentes au sommet de la grille.
3. Le jeu continue jusqu’à ce qu’un des joueurs ait un alignement continu de quatre pions de sa couleur. L’alignement peut être vertical, horizontal ou en diagonale.
4. Pour vider la grille, poussez la barre de retenue qui se trouve au vase de celle-ci et les pions tomberont. Vous pouvez maintenant commencer la partie suivante.

**Conseil :** Lire attentivement les questions suivantes et ébaucher une solution au brouillon avant de commencer la rédaction.

**Exercice 1 : Préparation du jeu**

Le jeu « puissance 4 » se joue sur une grille verticale de six lignes et sept colonnes avec 21 pions rouges et 21 pions jaunes.

Pour faciliter les calculs ultérieurs, nous allons ajouter une bordure à cette grille. Ses dimensions sont donc de huit lignes et neuf colonnes. Bien entendu, ces cases supplémentaires ne peuvent pas contenir de pions (qu'ils soient jaunes ou rouges). Elles sont donc toujours vides.

Définir les types nécessaires pour représenter la grille de jeu.

```

1  ! Définir le type pour une case de la grille
2
3  INTEGER, PARAMETER, PUBLIC :: VIDE=0
4  INTEGER, PARAMETER, PUBLIC :: ROUGE=1
5  INTEGER, PARAMETER, PUBLIC :: JAUNE=2
6
7  ! La grille de jeu avec une bordure supplémentaire
8  TYPE, PRIVATE :: typ_grille
9      INTEGER, DIMENSION(0:NB_LIGNES+1,0:NB_COLONNES+1)::grille
10 END TYPE typ_grille

```

**Exercice 2 : Vider la grille**

Pour commencer chaque partie, il faut commencer par vider la grille des jetons qu'elle contient. Écrire un sous-programme qui réalise cette opération.

**Exercice 3 : Afficher la grille**

Écrire un sous-programme qui affiche le contenu de la grille. Par convention, les jetons de couleur rouge seront représentés par un astérisque (\*) et les jetons jaunes par la lettre o minuscule.

Pour faciliter, la localisation d'une case de la grille, les numéros de lignes et de colonnes seront affichés comme sur la figure 1.

```

      1 2 3 4 5 6 7
6      *      6
5      o      5
4      o o *  4
3      * o o o 3
2      * o * o * 2
1 * o * * o o * 1
      1 2 3 4 5 6 7

```

FIGURE 1 – Affichage de la grille

**Exercice 4 : Décider si un coup est possible**

Écrire un sous-programme qui indique s'il est possible de jouer dans une colonne donnée.

**Exemple :** Sur l'exemple de la figure 1, il est possible de jouer les colonnes 1, 2, 3, 5, 6 ou 7. Il n'est pas possible de jouer la colonne 4.

**Exercice 5 : Lâcher un jeton**

Lorsqu'un joueur joue, il lâche un jeton de sa couleur au dessus d'une colonne de la grille. En

raison de la force gravitationnelle, le jeton tombe dans la grille jusqu'à ce qu'il soit arrêté par un autre jeton ou le fond de la grille.

Écrire un sous-programme qui réalise cette opération : il calcule le nouvel état de la grille lorsqu'un jeton est lâché au dessus d'une colonne donnée.

**Exemple :** Partant de la figure 1, si les rouges jouent la colonne 5, le nouvel état de la grille est celui donné par la figure 2.

	1	2	3	4	5	6	7	
6				*				6
5				o				5
4			o	o		*		4
3			*	o		o	o	3
2		*	o	*	*	o	*	2
1	*	o	*	*	o	o	*	1
	1	2	3	4	5	6	7	

FIGURE 2 – Les rouges ont joué la colonne 5

### Exercice 6 : Compter les jetons alignés

Écrire un sous-programme qui indique quelle est la longueur de l'alignement le plus long qui inclut un jeton donné (repéré par son numéro de colonne et son numéro de ligne). Les alignements doivent être cherchés dans toutes les directions (horizontale, verticale ou en diagonale).

Ce sous-programme sera ensuite utilisé pour déterminer la fin d'une partie (alignement  $\geq 4$ ) ou aider l'ordinateur à choisir la colonne où jouer (exercice 8).

Sur l'exemple de la figure 2, la case (1,1) correspond à un alignement de trois jetons en diagonale ; la case (7,1) correspond à un alignement de deux jetons verticalement ; la case (6,2) correspond à trois jetons alignés verticalement ou en diagonale...

**Indication :** Quel est l'intérêt d'avoir défini une bordure supplémentaire autour de la grille ?

### Exercice 7 : Conseiller une colonne où jouer

Écrire un sous-programme qui étant donné une grille propose une colonne où jouer. Le numéro de la colonne sera choisi aléatoirement.

### Exercice 8 : Améliorer le conseil

Pour améliorer le conseil, on ajoute une contrainte supplémentaire : conseiller la colonne qui permet l'alignement le plus long.

**Remarque :** Cette contrainte implique que si un coup gagnant existe, la colonne conseillée conduira à la victoire.

**Exemple :** Partant de la figure 2, les jaunes joueront en colonne 5. Ceci provoque un alignement de 4 pions et donc la victoire.