

UE NFP 136 (VARI 2)

Séances de TP 3 à 5 (suite)

Exercice supplémentaire : Puissance 4

Dans cet exercice, on se propose d’implémenter une version JAVA du célèbre jeu de Puissance 4, à l’aide d’un affichage en mode texte et d’une interface simple au clavier. On rappelle que ce jeu se joue à 2 joueurs, sur une grille verticale constituée de 6 lignes et 7 colonnes.

À tour de rôle, chacun des 2 joueurs va choisir une des 7 colonnes pour y jouer un de ses jetons. À chaque fois qu’un jeton est joué dans une colonne, il “descend” jusqu’à l’emplacement libre situé le plus bas sur cette colonne (pour simuler un jeton qui tomberait sous l’effet de la pesanteur), et cet emplacement devient alors occupé (par ce jeton). Une colonne est remplie lorsque plus aucun emplacement n’y est disponible, ce qui se produit lorsque 6 jetons ont été joués dans cette colonne. Lorsqu’une colonne est remplie, plus aucun joueur ne peut jouer de jeton dedans.

Dès qu’un des 2 joueurs arrive à aligner 4 de ses jetons (horizontalement, verticalement, ou en diagonale), il gagne aussitôt la partie. Si toutes les colonnes sont remplies mais qu’aucun des 2 joueurs n’a réussi à aligner 4 jetons, alors la partie se conclut sur un match nul.

Pour que le programme puisse écrire l’identité du vainqueur dans le cas où la partie ne se solderait pas par un match nul, on fournira en paramètres, au moment de l’appel du programme, les prénoms des 2 joueurs.

On se propose de modéliser la grille de jeu à l’aide d’une variable de type matrice (tableau à 2 dimensions) : ainsi, chaque emplacement de la grille correspondra à une case de cette matrice. Une case pourra donc soit être libre, soit être occupée par un jeton du joueur 1, soit être occupée par un jeton du joueur 2. Voici quelques indications sur la marche à suivre :

1. On écrira une méthode permettant d'afficher la grille : on affichera, en haut, les numéros des 7 colonnes, puis les éléments de chaque colonne les uns en-dessous des autres. Un jeton du joueur 1 sera représenté par un caractère donné (par exemple, 'X'), un jeton du joueur 2 sera représenté par un autre caractère (par exemple, 'O'), et un emplacement libre sera représenté par un troisième caractère (par exemple, '_'). On fournit ici, à titre indicatif, un exemple d'affichage d'une partie en cours où le joueur 1 a joué 2 jetons dans la colonne 1 et où le joueur 2 a joué un jeton dans la colonne 5 :

```

1234567
-----
-----
-----
-----
X-----
X__O__

```

2. On écrira une méthode permettant de tester si un des joueurs (le joueur courant) a réussi à aligner 4 jetons sur la grille (horizontalement, verticalement ou en diagonale). Il y a pour cela plusieurs façons de faire. L'une d'entre elles consiste à tester, pour chaque case de la matrice contenant un jeton du joueur, la présence consécutive de 4 jetons lui appartenant (dans les trois directions possibles) à partir de cette case.
3. On aura également besoin d'écrire une méthode gérant un tour de jeu, c'est-à-dire la saisie du numéro de colonne choisie pour jouer le jeton, puis l'ajout d'un jeton sur la case libre la plus basse de cette colonne (après avoir vérifié que le numéro de colonne est valide et que la colonne n'est pas pleine, sinon le joueur doit en choisir une autre !).
4. Enfin, on aura besoin de gérer l'enchaînement des tours de jeu (c'est-à-dire l'alternance entre les deux joueurs), ainsi que la fin de partie : après avoir joué son jeton, le joueur courant a-t-il remporté la partie, ou provoqué un match nul ?