

ReadMe Puissance 4

Pierre Cerantola, Reda Eloualladi, Clara Nguyen

Pr. Bartholomew George

Table des matières

Table des matières	1
1) Conventions :	2
2) Fonctionnalités particulières :	2
3) Prototypes des fonctions essentielles :	2

1) Conventions :

Type d'objet "plateau"

Dans tout le code, un plateau est du type "Plateau", qui est en réalité le type "Array[Array[Int]]".

Représentation des jetons dans un plateau de type Plateau

- l'assertion "avoir un jeton jaune à la colonne i et à la ligne j" est équivalente à `"plateau(i)(j) == 1"`
- l'assertion "avoir un jeton rouge à la colonne i et à la ligne j" est équivalente à `"plateau(i)(j) == 2"`
- l'assertion "avoir une case vide à la colonne i et à la ligne j" est équivalente à `"plateau(i)(j) == 0"`

Indices

Le comptage de l'indice de la ligne s'effectue en décroissant. Par exemple, dans un plateau standard de taille 7 colonnes * 6 lignes, `"plateau(6)(0)"` désigne la case extrême Nord-Est du plateau, et `"plateau(6)(5)"` désigne la case extrême Sud-Est du plateau.

Note : dans certaines fonctions, on utilise la transposée de cette représentation, grâce à la fonction `inverse(plateau)`.

Couleur des joueurs

- le joueur qui joue jaune porte le numéro 1
- le joueur qui joue rouge porte le numéro 2

Représentation des jetons lors des entrées-sorties à la console :

- un jeton jaune est symbolisé par "J"

- un jeton rouge est symbolisé par "R"
- une case vide est symbolisée par "."

2) Fonctionnalités particulières :

Tout le code est généralisé sur des grilles de tailles $n \times m$.

On a la possibilité de choisir contre quelle IA on veut jouer, et on peut se faire affronter 2 IA.

3) Prototypes des fonctions essentielles :

saisieEntier : Int

// fonction qui demande à l'utilisateur de saisir un entier tant que sa saisie est incorrecte. Renvoie ensuite cet entier

saisieNbColonnes(): Int

// fonction qui renvoie le nombre de colonnes saisi par l'utilisateur si et seulement celui-ci est strictement supérieur à 6. Redemande à l'utilisateur sinon

saisieNbLignes(): Int

// fonction qui renvoie le nombre de lignes saisi par l'utilisateur si et seulement celui-ci est strictement supérieur à 6. Redemande à l'utilisateur sinon

remplaceElemPlateau(plateau: Plateau, numColonne: Int, numLigne: Int, element: Int): Plateau

// fonction qui renvoie le plateau dont l'élément "plateau(numColonne)(numLigne)" a été remplacé par "element"

importPlateau(chaine: String): Plateau

// fonction qui renvoie un plateau construit à partir d'une chaîne de caractères "chaine". Dans la "chaine", le séparateur des lignes du plateau à construire doit être ' ', les jetons doivent être 'R' pour rouge et 'J' pour jaune, et les cases vides doivent être '.'

// exemple d'utilisation : "import(RJJJRJR JRJRRJR JRJRJJ .R.R... ..)" renvoie le plateau de la page 2 de l'énoncé du projet

exportPlateau(plateau: Plateau): String

// fonction qui renvoie une chaîne de caractères contenant toutes les lignes du plateau "plateau"

// exemple d'utilisation : "exportPlateau(plateau)" renvoie la chaîne "RJJJRJR JRJRRJR JRJRJJ .R.R... .." si "plateau" est le plateau de la page 2 de l'énoncé du projet

constrChaineNumColsRT(chaine: String, numColonne: Int, nbColonnes: Int): String

// fonction qui renvoie la chaîne des numéros de colonne du plateau "plateau"

afficherPlateau(plateau: Plateau): Unit

// procédure qui affiche le plateau "plateau" dans la console

// exemple d'utilisation : "afficherPlateau(plateau)" (avec le "plateau" de la page 2 de l'énoncé du TP) affichera en console :

```
0 1 2 3 4 5 6
. . . . . .
. . . . . .
. R . R . . .
J R J R J J .
J R J R R J R
R J J J R J R
```

jouerColonne(plateau: Plateau, numColonne: Int, couleurJoueur: Int): Boolean

// procédure qui renvoie "true" si la colonne "numColonne" n'est pas remplie, et dans ce cas elle place un jeton de couleur "couleurJoueur" dans la colonne numéro "numColonne" du plateau "plateau". Si la colonne "numColonne" est remplie ou dépasse les limites du plateau, la fonction renvoie "false" et ne touche pas au plateau

joueCoupHumain(plateau: Plateau, numColonne: Int) : Unit

// procédure destinée au joueur humain qui lui fait jouer 1 jeton de couleur 1 dans la colonne "numColonne" du plateau "plateau". Si la colonne "numColonne" n'est pas disponible, la fonction avertit l'utilisateur à la console, lui redemande une nouvelle colonne puis boucle sur elle-même. NOTE : la fonction "jouerColonne()" est utilisée dans cette fonction

verifsaisieintervalle(debut: Int, fin : Int, saisie :Int) : Int

//fonction qui vérifie si l'entier saisie est dans l'intervalle debut fin

valeurFenetre(fenetre : Array[Int], couleur: Int) : Int

//fonction qui attribut un score à une configuration fenetre

estDispoColonne(plateau: Plateau, numColonne: Int): Boolean

// fonction qui renvoie "true" si la colonne "numColonne" est dispo et "false" sinon

listerColsDispo(plateau: Plateau): Array[Int]

// fonction qui renvoie le tableau des indices des colonnes disponibles dans le plateau "plateau"

**minMax(plateau: Plateau, couleurJoueurRef: Int, couleurJoueur:
Int, profondeur: Int): Int**

// fonction qui renvoie la valeur du meilleur coup à jouer pour le joueur jouant la couleur
“couleurJoueurRef”, à partir du plateau “plateau”

iaCoup(plateau: Plateau, couleurJoueur: Int, profondeur: Int): Int

// fonction qui renvoie l'indice de la meilleure colonne à jouer sur le coup suivant pour le joueur
jouant la couleur “couleurJoueur”, sur le plateau “plateau”

inverse(plateau : Plateau): Plateau

//fonction qui retourne un plateau (transpose les lignes et les colonnes)

valeurCoupHor(plateauVirt: Plateau, couleur: Int) : Int

//fonction qui attribue un score à un plateau en fonction des points alignés horizontalement sur le plateau

valeurCoupVert(plateauVirt: Plateau, couleur: Int) : Int

//fonction qui attribue un score à un plateau en fonction des points alignés verticalement sur le plateau

valeurCoupDiag(plateauVirt: Plateau, couleur: Int) : Int

//fonction qui attribue un score à un plateau en fonction des points alignés en diagonale sur le plateau

//Ces trois fonctions permettent de calculer la valeur d'un coup (d'un plateau) en parcourant le tableau avec toutes les fenêtres de 4 cases possibles.

Une fenêtre est un Array[Int] représentant 4 cases consécutives formant une ligne horizontale, verticale ou diagonale, respectivement pour chaque fonction.

Elle renvoie la somme des valeurs des fenêtres du type correspondant (horizontales, verticales ou diagonales).

valeurCoup(plateauVirt: Plateau, couleur: Int): Int

// fonction qui attribue un score globale à un plateau donné

//Renvoie la valeur d'un coup en additionnant le résultat de trois fonctions : ValeurCoupHor, ValeurCoupVert et ValeurCoupDiag

meilleurCoup(plateau: Plateau, couleur: Int) : Int

// fonction qui renvoie l'indice de la meilleure colonne à jouer pour le joueur de couleur "couleur" et dans le plateau "plateau"

// Choisir le meilleur coup à jouer en essayant chaque coup possible dans un plateau virtuel et en calculant sa valeur avec la fonction valeurCoup()

plateauPlein(plateau: Plateau): Boolean

// fonction qui renvoie "true" si le plateau "plateau" est plein et "false" sinon

aGagneVertical(plateau: Plateau, couleurJoueur: Int): Boolean

// fonction qui renvoie "true" si 4 jetons de couleur "couleurJoueur" sont alignés verticalement dans une des colonnes du plateau "plateau" et "false" sinon

aGagneHorizontal(plateau: Plateau, couleurJoueur: Int): Boolean

// fonction qui renvoie "true" si 4 jetons de couleur "couleurJoueur" sont alignés horizontalement dans une des lignes du plateau "plateau" et "false" sinon

aGagneDiagAscVert(plateau: Plateau, couleurJoueur: Int): Boolean

// fonction qui renvoie "true" si 4 jetons de couleur "couleurJoueur" sont alignés sur une diagonale ascendante verticale du plateau "plateau" et "false" sinon

aGagneDiagDescVert(plateau: Plateau, couleurJoueur: Int):

Boolean

// fonction qui renvoie "true" si 4 jetons de couleur "couleurJoueur" sont alignés sur une diagonale descendante verticale du plateau "plateau" et "false" sinon

aGagne(plateau: Plateau, couleurJoueur: Int) : Boolean

// fonction qui renvoie "true" si le joueur de couleur "couleurJoueur" a gagné la partie sur le plateau "plateau"

// cette fonction détecte un alignement de 4 jetons sur le plateau en utilisant les 4 fonctions aGagneHorizontal(), aGagneVertical(), aGagneDiagAscVert() et aGagneDiagDescVert()

jouer(plateau : Plateau, couleurJoueur : Int, typ : Int, diff : Int):

Unit

//procédure qui joue un coup dans le plateau en fonction du type de joueur

partie(plateau : Plateau) : Unit

//procédure qui implémente une parti de puissance4

Rappel : la fonction jouer remplace la fonction joueCoupOrdi car nous avons plusieurs IA

joueCoupOrdi(plateau: Plateau, couleur:Int) : Unit

// Joue le coup de l'ordinateur en fonction de l'IA choisie : soit RANDOM, soit MEILLEURCOUP