Justine ******, 1D2

BUT Informatique, semestre 1
IUT de Lannion

Année 2022-2023

~ S.A.É. 1.01 - Implémentation d'un besoin client ~

PUISSANCE 4

{Maquette}

{Décomposition algorithmique}

{Cahier de tests}

{Code source intermédiaire}

{Code source final}

{Traces d'exécution des tests}

Test de la procédure *initGrille()*

IDENTIFICATION DU TEST Initialise la grille en affectant la constante VIDE à chacun de ses éléments

CONTEXTE D'EXÉCUTION Exécution de la procédure complète

ÉTAPES DE TEST À EFFECTUER Lancer la procédure initGrille()

N° CAS	DONNÉES DU TEST	RÉSULTATS ATTENDUS	RÉSULTATS OBTENUS	RÉSULTAT DU TEST
Cas 1	La grille n'est pas initialisée	La grille est initialisée	La grille est initialisée	✅
Cas 2	La grille est déjà initialisée	La grille est initialisée	La grille est initialisée	✅
Cas 3	La grille est déjà remplie avec des jetons	La grille est initialisée	La grille est initialisée	✅

Traces d'exécution des tests

Testé dans le fichier test_initgrille.c (joint)



Cas 1 : OK



Cas 2 : OK



Cas 3 : OK

Test de la procédure *afficher()*

IDENTIFICATION DU TEST Affiche la grille actualisée et le prochain pion à placer

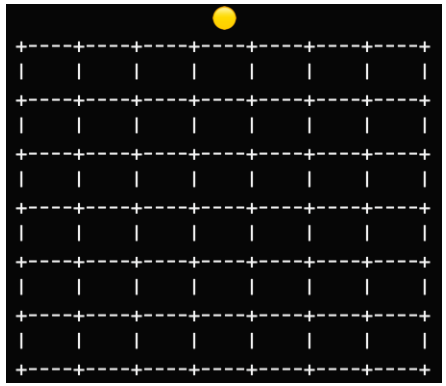
CONTEXTE D'EXÉCUTION Exécution de la fonction complète

ÉTAPES DE TEST À EFFECTUER Lancer la procédure afficher()

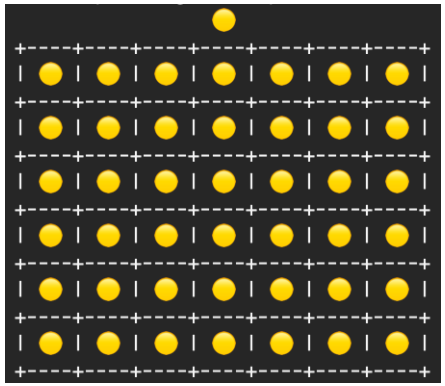
N° CAS	DONNÉES DU TEST	RÉSULTATS ATTENDUS	RÉSULTATS OBTENUS	RÉSULTAT DU TEST
Cas 1	La grille est vide	La grille est affichée sans pion	La grille est affichée sans pion	✅
Cas 2	La grille est pleine	La grille est affichée pleine	La grille est affichée pleine	✅
Cas 3	La grille est partiellement remplie (en cours de partie)	La grille est affichée telle quelle	La grille est affichée telle quelle	✅

Traces d'exécution des tests

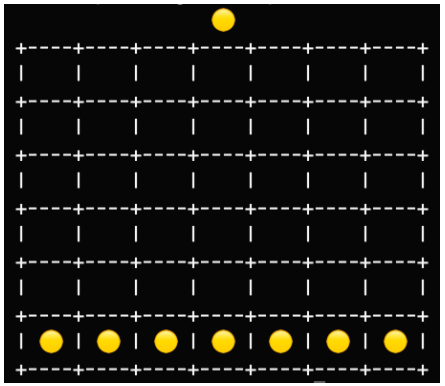
Testé dans le fichier test_afficher.c (joint)



Cas 1 : OK



Cas 2 : OK



Cas 3 : OK

Test de la fonction *grillePleine()*

IDENTIFICATION DU TEST Teste si toutes les cases de la grille sont occupées

CONTEXTE D'EXÉCUTION Exécution de la fonction complète

ÉTAPES DE TEST À EFFECTUER Lancer la fonction grillePleine()

N° CAS	DONNÉES DU TEST	RÉSULTATS ATTENDUS	RÉSULTATS OBTENUS	RÉSULTAT DU TEST
Cas 1	La grille n'est pas pleine	false (0)	false (0)	✅
Cas 2	La grille n'est pas pleine mais la ligne supérieure est remplie*	ERREUR	false (0)	❌
Cas 3	La grille est pleine (42 jetons)	true (1)	false (1)	✅

** Pas sûre d'avoir compris celle-ci…* 😕

Traces d'exécution des tests

Testé dans le fichier test_grillepleine.c (joint)

(J'ai ajouté une condition pour que les 0 affichent "false" et les 1 "true")



Cas 1 : OK



Cas 2 : Pas OK



Cas 3 : OK

Test de la procédure *jouer()*

IDENTIFICATION DU TEST Appelle la fonction choisirColonne() puis la fonction trouverLigne()

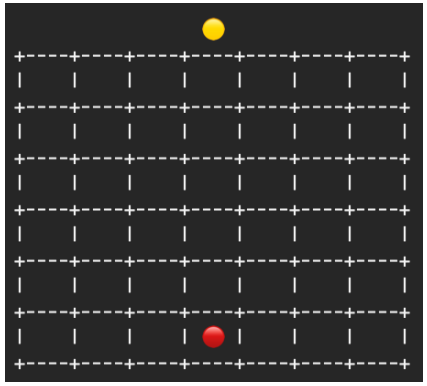
CONTEXTE D'EXÉCUTION Exécution de la procédure complète

ÉTAPES DE TEST À EFFECTUER Vérifier la valeur renvoyée par trouverLigne()

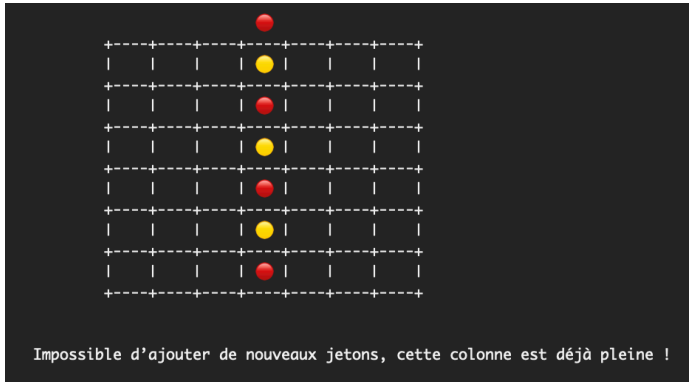
N° CAS	DONNÉES DU TEST	RÉSULTATS ATTENDUS	RÉSULTATS OBTENUS	RÉSULTAT DU TEST
Cas 1	On joue dans une colonne non pleine	La pièce est placée dans la cellule correspondante	La pièce est placée dans la cellule correspondante	✅
Cas 2	On joue dans une colonne pleine	message d'avertissement : colonne pleine puis relance choisirColonne()	message d'avertissement : colonne pleine puis relance choisirColonne()	✅

Traces d'exécution des tests

Testé directement dans le fichier puissance4.c



Cas 1 : OK
Le joueur rouge a joué sa pièce en 4



Cas 2 : OK
Le joueur rouge essaie de placer son jeton en 4 : le message d'erreur s'affiche puis le programme demande une nouvelle saisie

Test de la fonction *choisirColonne()*

IDENTIFICATION DU TEST Placer son jeton dans une colonne

CONTEXTE D'EXÉCUTION Exécution de la fonction complète

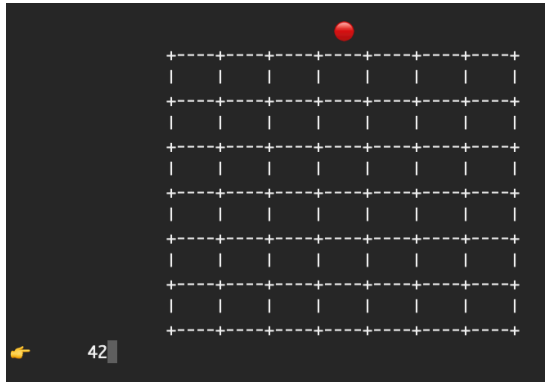
ÉTAPES DE TEST À EFFECTUER Saisie d'une ou plusieurs valeurs à la suite

N° CAS	DONNÉES DU TEST	RÉSULTATS ATTENDUS	RÉSULTATS OBTENUS	RÉSULTAT DU TEST
Cas 1	Saisie d'un caractère non valide pour le déplacement	Redemande la saisie*	Redemande la saisie	✅ (?)
Cas 2	Saisir [q]	La pièce se déplace d'une colonne vers la gauche	La pièce se déplace d'une colonne vers la gauche	✅
Cas 3	Saisir [d]	La pièce se déplace d'une colonne vers la droite	La pièce se déplace d'une colonne vers la droite	✅
Cas 4	Saisir [q] à la borne gauche	Message d'erreur : sortie de grille	Message d'erreur : sortie de grille	✅
Cas 5	Saisir [d] à la borne droite	Message d'erreur : sortie de grille	Message d'erreur : sortie de grille	✅
Cas 6	Saisir [espace] dans une colonne valide	La pièce est placée dans la colonne sélectionnée	La pièce est placée dans la colonne sélectionnée	✅

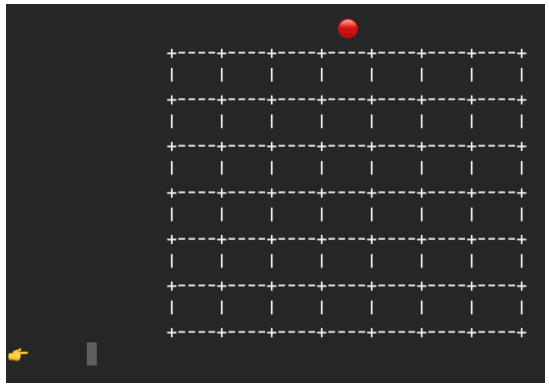
* *Bon, initialement, je voulais qu'un message d'erreur s'affiche avant de redemander la saisie à l'utilisateur, mais pour une raison obscure, ça ne veut pas fonctionner… Donc certes, ça finit bien par redemander la saisie, mais comme ce n'était pas tout à fait le résultat escompté, pas sûre que le test soit validé…* 😶

Traces d'exécution des tests

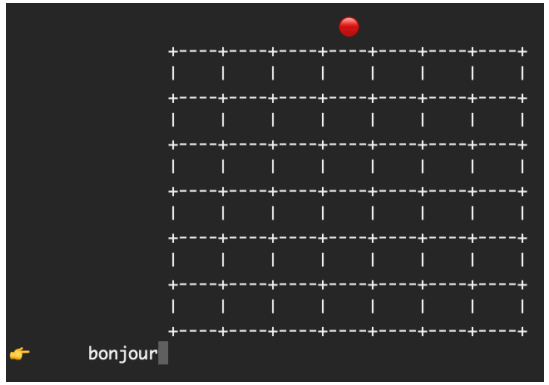
Testé directement dans le fichier puissance4.c



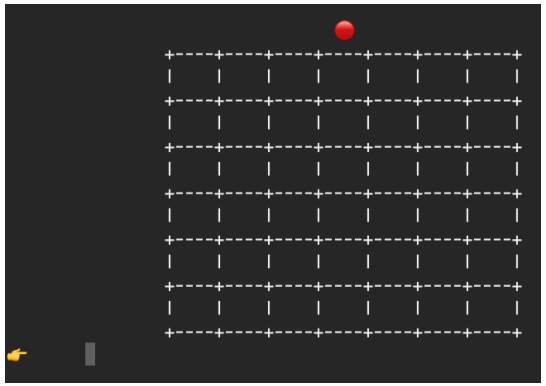
Cas 1 (1)
Le joueur saisit un nombre



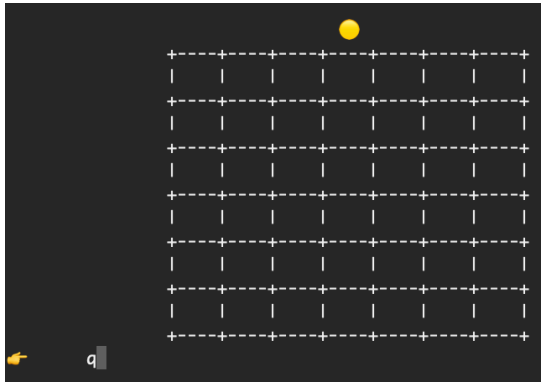
Cas 1 (2) : OK
Après validation, une nouvelle saisie est demandée



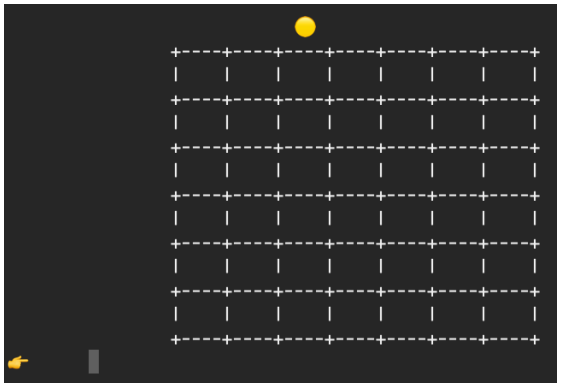
Cas 1 bis (1)
Le joueur, ce petit polisson, saisit un mot, cette fois-ci



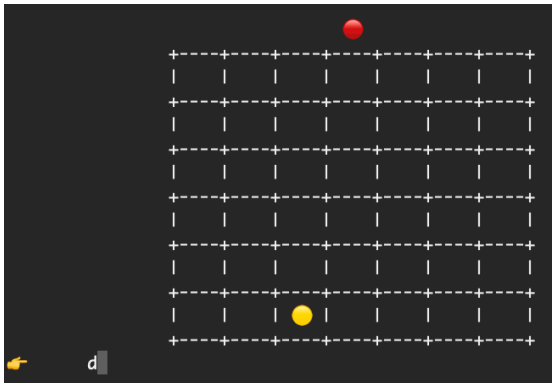
Cas 1 bis (2) : OK
Pas de chance, la programmeuse a tout prévu (ou pas), une nouvelle saisie est encore une fois demandée



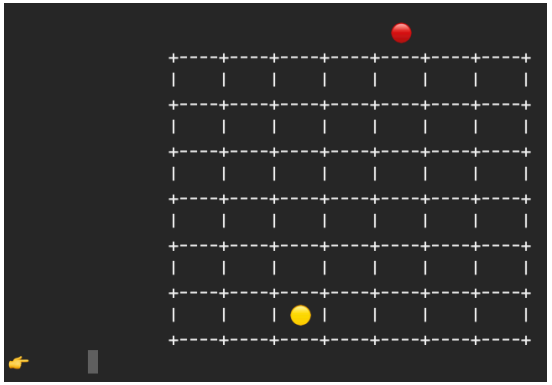
Cas 2 (1)
Le joueur a saisi [q]



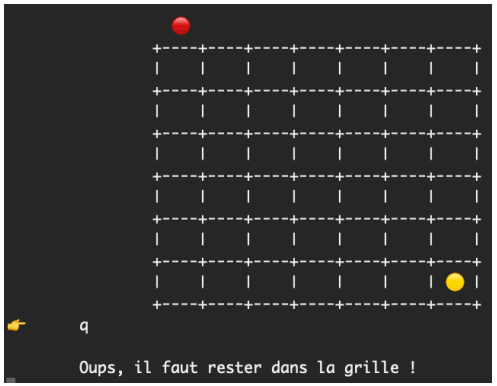
Cas 2 (2) : OK
Après validation, le pion s'est bien déplacé vers la gauche



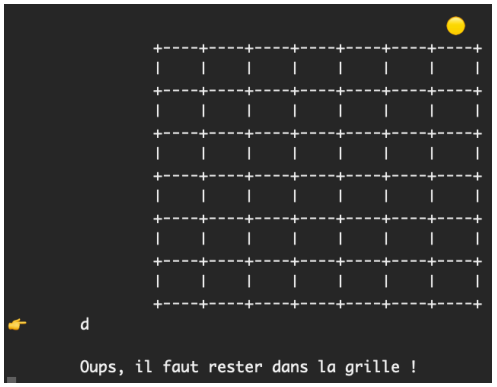
Cas 3 (1)
Le joueur a saisi [d]



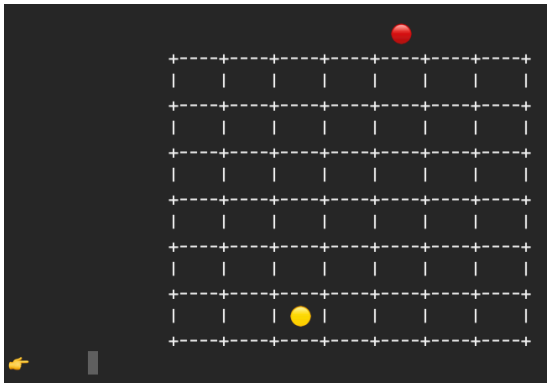
Cas 3 (2) : OK
Après validation, le pion s'est bien déplacé vers la droite



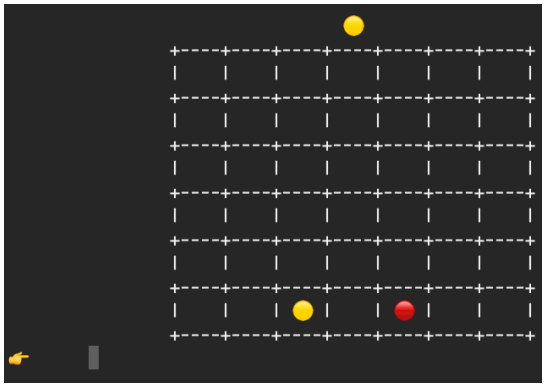
Cas 4 : OK
Le joueur essaie de sortir de la grille par la gauche (raté)



Cas 5 : OK
Le joueur essaie de sortir de la grille par la droite (encore raté)



Cas 6 (1)
Le joueur a saisi [espace]



Cas 6 (2) : OK
Après validation, le pion est bien tombé dans la colonne choisie (GG)

Test de la fonction *trouverLigne()*

IDENTIFICATION DU TEST Parcourt la colonne sélectionnée en quête de la première case vide

CONTEXTE D'EXÉCUTION Exécution de la procédure trouverLigne()

ÉTAPES DE TEST À EFFECTUER Lancer la procédure trouverLigne()

N° CAS	DONNÉES DU TEST	RÉSULTATS ATTENDUS	RÉSULTATS OBTENUS	RÉSULTAT DU TEST
Cas 1	La colonne est pleine	-1	-1	✅
Cas 2	La colonne n'est pas pleine	Le numéro de la ligne correspond à la première case libre de la colonne	Le numéro de la ligne correspond à la première case libre de la colonne	✅

Traces d'exécution des tests

Testé dans le fichier test_trouverligne.c (joint)



Cas 1 : OK



Cas 2 : OK
La colonne était vide, la fonction renvoie donc 5 (indice de la dernière ligne)

Test de la fonction *estVainqueur()*

IDENTIFICATION DU TEST Détermine si un alignement gagnant a été fait

CONTEXTE D'EXÉCUTION Exécution de la fonction complète

ÉTAPES DE TEST À EFFECTUER Lancer la fonction estVainqueur()

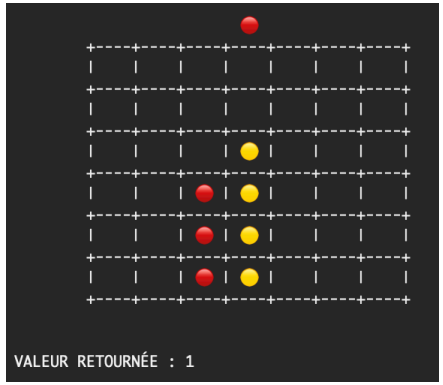
N° CAS	DONNÉES DU TEST	RÉSULTATS ATTENDUS	RÉSULTATS OBTENUS	RÉSULTAT DU TEST
Cas 1	Alignement gagnant vertical	true (1)	true (1)	✅
Cas 2	Alignement gagnant horizonal	true (1)	true (1)	✅
Cas 3	Alignement gagnant diagonal ↗	true (1)	true (1)	✅
Cas 4	Alignement gagnant diagonal ↘	true (1)	true (1)	✅
Cas 5	Grille pleine, mais aucun alignement	false (0)	false (0)	✅

* *Pas compris le cas "ligne d'au moins 4 pions interrompue par un pion adverse"* 😐

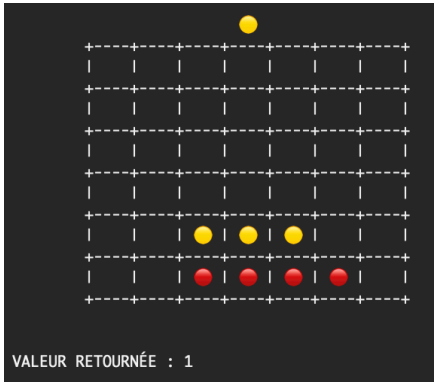
Traces d'exécution des tests

Testé directement dans le fichier puissance4.c en ajoutant dans la fonction estVainqueur() la ligne :

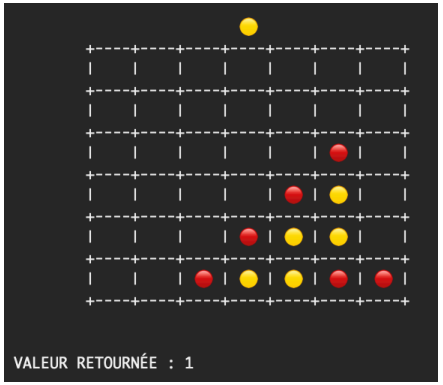
```
printf("\tVALEUR RETOURNÉE : %d\n", resultat_partie);
```



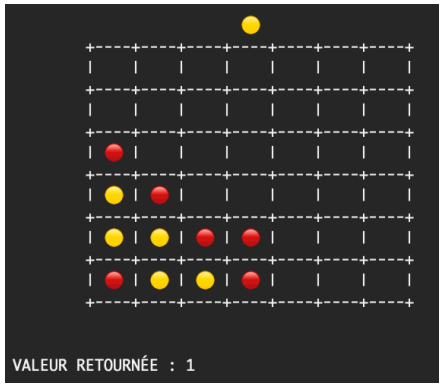
Cas 1 : OK



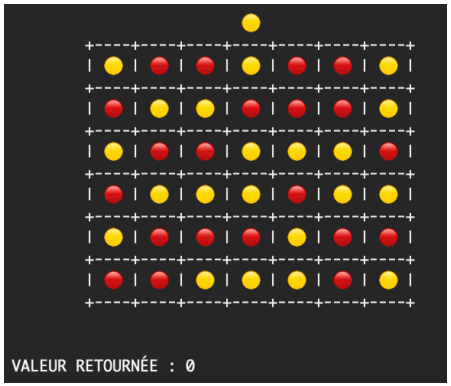
Cas 2 : OK



Cas 3 : OK



Cas 4 : OK



Cas 5 : OK

Test de la procédure *finDePartie()*

IDENTIFICATION DU TEST Affiche l'éventuel(le) gagnant(e)

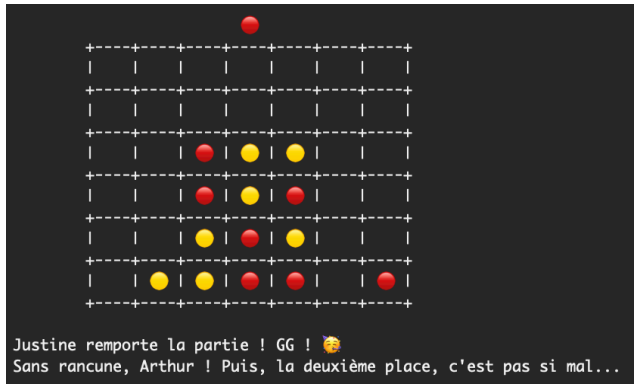
CONTEXTE D'EXÉCUTION Exécution de la procédure complète

ÉTAPES DE TEST À EFFECTUER Lancer la procédure finDePartie()

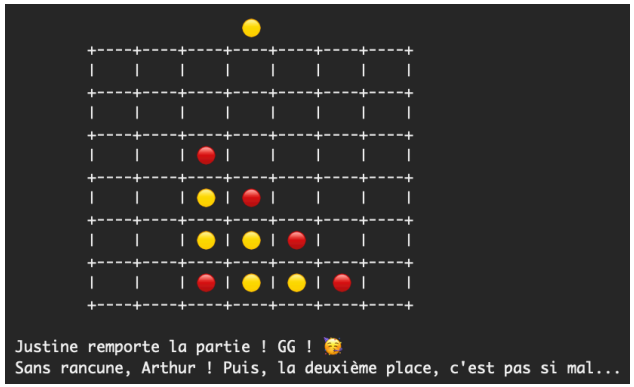
N° CAS	DONNÉES DU TEST	RÉSULTATS ATTENDUS	RÉSULTATS OBTENUS	RÉSULTAT DU TEST
Cas 1	Si le joueur jouant les pièces jaunes a gagné la partie	Message de félicitation pour le joueur jaune	Message de félicitation pour le joueur jaune	✅
Cas 2	Si le joueur jouant les pièces rouges a gagné la partie	Message de félicitation pour le joueur rouge	Message de félicitation pour le joueur rouge	✅
Cas 3	En cas de match nul	Message de félicitation aux deux joueurs	Message de félicitation aux deux joueurs	✅

Traces d'exécution des tests

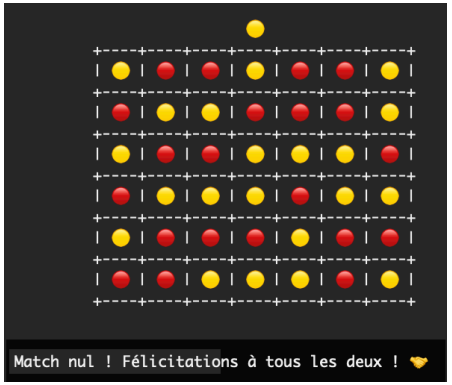
Testé directement dans le fichier puissance4.c



Cas 1 : OK
La joueuse aux pions jaunes a gagné



Cas 2 : OK
La joueuse aux pions rouges a (encore) gagné



Cas 3 : OK
Match nul : la joueuse des 2 premiers cas de test a probablement changé d'adversaire...