

Glossaire Tetros :

Légende :

V = Variable

F = Fonction

C = Constante

Les documentations ne sont présentes que sur les points les plus importants du code.

Toutes les variables ne sont pas présentées, uniquement les principales.

<u>Fichier</u>	<u>Type : Nom</u>	<u>Description</u>
tetros.c (Ligne 46)	V: mainGrid	Grille principale du jeu, on y place les tetriminos. Documentation
tetros.c (Ligne 47)	V: mobileGrid	Grille du tetrimino actuellement contrôlé par le joueur. Documentation
gameUI.c	F: drawGrid	Affiche une grille donnée dans la fenêtre de jeu
gameUI.c	F: drawGhostblocks	Récupère en argument mainGrid et mobileGrid pour afficher une prévisualisation de l'emplacement du tetrimino à la fin de la chute. Documentation
tetrimino.c (Ligne 275)	V: tempGrid	Grille temporaire utilisée dans diverses fonctions.
tetrimino.c	F: goBottom	Fonction qui fait descendre le bloc ligne par ligne (instantanément); jusqu'à rencontrer un autre bloc ou la dernière ligne de la grille. La fonction placer est alors appelée par cette même fonction. Documentation
regles.c (Ligne 8)	C: NBLINES	Nombre de lignes dans la grille de jeu, les 2 lignes invisibles au-dessus de la grille affichées sont aussi comptées. Documentation
regles.c (Ligne 9)	C: NBCOLUMNS	Nombre de colonnes dans la grille de jeu.
gameUI.c	F: pause	Met le jeu en pause et affiche une fenêtre par dessus la grille afin d'empêcher le joueur de tricher en réfléchissant.

tetros.c (Ligne 48)	V: inventory	Entier correspondant au bloc stocké dans l'inventaire.
tetros.c (Ligne 49)	V: invUsed	Booléen si l'inventaire a déjà été utilisé ou non.
tetrimino.c	F: initGrid	Initialise une grille donnée.
tetros.c (Ligne 55)	V: gridWindow	Fenêtre dans laquelle on affiche la grille.
tetros.c (Ligne 59)	V: tetriminoID	Entier affecté au tetrimino actuellement en jeu.
tetros.c (Ligne 60)	V: priorID	Entier affecté au tetrimino précédent en jeu.
tetros.c (Ligne 64)	V: key	Entier correspondant à une touche donnée et pressée par l'utilisateur.
tetros.c (Ligne 56)	V: inGame	Booléen -> tant que sa valeur est true, la partie actuelle est en cours.
tetros.c (Ligne 22)	V: gameOn	Booléen -> tant que sa valeur est true, on ne termine pas l'affichage ncurses et le programme. <u>Différent</u> de inGame
tetrimino.c	F: putTetrimino	Place le tetrimino de mobileGrid dans mainGrid. Documentation
tetrimino.c	F: turnTetrimino	Fonction de rotation. Tourne le tetrimino à 90° dans le sens horaire.
regles.c (Ligne 11)	C: DEBUG_MODE	Booléen, vrai si le mode debug est activé, faux sinon.
tetrimino.c	F: isGridfull F: isGridempty	Prend une grille en entrée et vérifie si le haut de la grille est atteint. Renvoie un booléen Prend une grille en entrée et vérifie si la grille est vide. Renvoie un booléen
tetrimino.c	F: setRandom	Génère un nombre aléatoire entre 1 et 8 en se basant sur rand() et la date.

tetrimino.c	F: genTetrimino	Prend en argument l'entier correspondant au tetrimino à générer ainsi qu'une grille pour ajouter dans celle-ci ce tétrimino.
tetrimino.c	F: goDown	Prend en argument mainGrid, mobileGrid ainsi que blockID et déplace le tétrimino vers le bas si possible, sinon appelle la fonction placer.
tetrimino.c	F: goLeft	Déplace le tétrimino à gauche si possible.
tetrimino.c	F: goRight	Déplace le tétrimino à droite si possible.
tetrimino.c (Lignes 11-18)	C: BLOCK_X	Constante (ici un entier) associée à chaque tetrimino.
debug.c	F: drawDebug	Affiche les valeurs internes de la matrice utilisées par le jeu.
debug.c	F: genDebugtetrimino	Génère un tétrimino avec 9 valeurs différentes.
tetrimino.c	F: downGridfromline	Descend d'une ligne l'ensemble de la grille donnée en entrée.
gameUI.c	F: drawScore	Affiche le score actuel du joueur en bas à côté de l'arène de jeu du joueur
gameUI.c	F: drawInventory	Montre au joueur le contenu de l'inventaire (si non vide). Donne la couleur du tetrimino et une lettre le représentant.
gameUI.c	F: drawCommands	Affiche les commandes à l'utilisateur à côté du jeu
gameUI.c	F: blinkLine	Affiche un clignotement blanc lorsqu'une ligne est complétée
gameUI.c	F: drawDifficulty	Affiche le niveau de difficulté sélectionné par le joueur

regles.c	F: setDifficulty	Définit le temps de référence du jeu en fonction de la difficulté
animation.c	F: animatedTitle	contient l'ensemble des frames animées contenant l'animation de lancement.
animation.c	F: startAnimation	Prépare l'animation lors du lancement du programme
animation.c	F: exitAnimation	Prépare l'animation lors de la sortie du programme
gameUI.c	F: drawGameMode	Affiche durant la partie le mode de jeu sélectionné
gameUI.c	F: drawTimer	Affiche le temps passé depuis le lancement du programme
mainGame.c	F: playTetros	Permet de lancer le jeu Tetros après l'apparition du menu (d'où le fait qu'il soit à côté de main.c)
mainMenu.c	F: initNcurses	Lance le module ncurses ainsi que les couleurs
mainMenu.c	F: drawDifficultyPicker	Affiche la fenêtre ncurses contenant les difficultés proposées au joueur
	F: difficultyPicker	Affiche le menu de sélection de la difficulté
mainMenu.c	F: gameModePicker	Affiche le menu permettant la sélection du mode de jeu
	F: drawGameModePicker	Affiche la fenêtre contenant les modes de jeux
mainMenu.c	F: drawButton	Affiche les boutons principaux de sélection lors du lancement du menu
mainMenu.c	F: drawMenu	Affiche le menu principal
mainMenu.c	F: drawMenuscore	Affiche les meilleurs scores et le dernier score en mémoire
mainMenu.c	F: drawHelp	Affiche le menu d'aide. L'adresse mail est vraie et

		est constituée de la première lettre de nos noms de famille (Balmès, Court et Frachet). Le jeu a été rendu public via Github
mainMenu.c	F: drawCredits	Affiche le menu complet des crédits dans lesquels se situent nos noms
rules.c	F: scoring	Calcule le nombre de points à amener dans le score
rules.c	F: switchInventory	Échange le tetrimino en cours et celui de l'inventaire
rules.c	F: timer	Calcule le nombre de temps passé depuis le lancement du jeu
tetrimino.c	F: isDownfree	Vérifie si le bas du tetrimino est libre, càd s'il n'est pas au-dessus d'un bloc
tetrimino.c	F: countLinesfull	Vérifie si l'une des grilles de la grille est pleine