

## Puissance N

C. Allaire, L. Bouriel et M. Burger

Généré par Doxygen 1.9.1



<b>1 Index des structures de données</b>	<b>1</b>
1.1 Structures de données	1
<b>2 Index des fichiers</b>	<b>3</b>
2.1 Liste des fichiers	3
<b>3 Documentation des structures de données</b>	<b>5</b>
3.1 Référence de la structure Grid	5
3.1.1 Description détaillée	5
3.1.2 Documentation des champs	5
3.1.2.1 grille	5
3.1.2.2 largeur	6
3.1.2.3 longueur	6
3.2 Référence de la structure Joueur	6
3.2.1 Description détaillée	6
3.2.2 Documentation des champs	6
3.2.2.1 couleur	6
3.2.2.2 initiative	7
3.2.2.3 jeton	7
3.3 Référence de la structure Origine	7
3.3.1 Description détaillée	7
3.3.2 Documentation des champs	7
3.3.2.1 abscisse	7
3.3.2.2 ordonnee	7
<b>4 Documentation des fichiers</b>	<b>9</b>
4.1 Référence du fichier ajout_retrait_jetons.c	9
4.1.1 Description détaillée	9
4.1.2 Documentation des fonctions	9
4.1.2.1 add_token()	9
4.1.2.2 colonne()	10
4.1.2.3 grille_pleine()	10
4.1.2.4 ligne_vide()	11
4.1.2.5 remove_token()	11
4.2 Référence du fichier ajout_retrait_jetons.h	11
4.2.1 Description détaillée	12
4.2.2 Documentation des fonctions	12
4.2.2.1 add_token()	12
4.2.2.2 colonne()	13
4.2.2.3 grille_pleine()	13
4.2.2.4 ligne_vide()	13
4.2.2.5 remove_token()	14
4.3 Référence du fichier check_winner.c	14

4.3.1 Description détaillée	15
4.3.2 Documentation des fonctions	15
4.3.2.1 check_diagonale()	15
4.3.2.2 check_diagonale_croissante()	15
4.3.2.3 check_diagonale_decroissante()	16
4.3.2.4 check_horizontal()	16
4.3.2.5 check_verticale()	17
4.3.2.6 check_winner()	17
4.3.2.7 origine_diagonale_croissante()	18
4.3.2.8 origine_diagonale_decroissante()	18
4.4 Référence du fichier check_winner.h	18
4.4.1 Description détaillée	19
4.4.2 Documentation des fonctions	19
4.4.2.1 check_diagonale()	19
4.4.2.2 check_diagonale_croissante()	20
4.4.2.3 check_diagonale_decroissante()	20
4.4.2.4 check_horizontal()	20
4.4.2.5 check_verticale()	22
4.4.2.6 check_winner()	22
4.4.2.7 origine_diagonale_croissante()	23
4.4.2.8 origine_diagonale_decroissante()	23
4.5 Référence du fichier gestion_grille.c	24
4.5.1 Description détaillée	24
4.5.2 Documentation des fonctions	24
4.5.2.1 choix_nbjeueur()	24
4.5.2.2 color()	24
4.5.2.3 free_memory()	25
4.5.2.4 initialisation()	25
4.5.2.5 load()	25
4.5.2.6 menu()	26
4.5.2.7 save()	26
4.5.2.8 selection_joueur()	26
4.5.2.9 show_grid()	27
4.6 Référence du fichier gestion_grille.h	27
4.6.1 Description détaillée	28
4.6.2 Documentation des fonctions	28
4.6.2.1 choix_nbjeueur()	28
4.6.2.2 color()	28
4.6.2.3 free_memory()	29
4.6.2.4 initialisation()	29
4.6.2.5 load()	29
4.6.2.6 menu()	30

4.6.2.7 save()	30
4.6.2.8 selection_joueur()	30
4.6.2.9 show_grid()	31
4.7 Référence du fichier header_principal.h	31
4.7.1 Description détaillée	32
4.8 Référence du fichier jeu.c	32
4.8.1 Description détaillée	32
4.8.2 Documentation des fonctions	32
4.8.2.1 affichage_tropee()	33
4.8.2.2 jeu_multi()	33
4.8.2.3 jeu_solo()	33
4.9 Référence du fichier jeu.h	33
4.9.1 Description détaillée	34
4.9.2 Documentation des fonctions	34
4.9.2.1 affichage_tropee()	34
4.9.2.2 jeu_multi()	34
4.9.2.3 jeu_solo()	35
4.10 Référence du fichier main.c	35
4.10.1 Description détaillée	35
4.10.2 Documentation des fonctions	36
4.10.2.1 main()	36
4.11 Référence du fichier ordinateur.c	36
4.11.1 Description détaillée	36
4.11.2 Documentation des fonctions	36
4.11.2.1 choix_action_ordi()	36
4.11.2.2 colonne_ordi()	37
4.12 Référence du fichier ordinateur.h	37
4.12.1 Description détaillée	37
4.12.2 Documentation des fonctions	38
4.12.2.1 choix_action_ordi()	38
4.12.2.2 colonne_ordi()	38
<b>Index</b>	<b>39</b>



# Chapitre 1

## Index des structures de données

### 1.1 Structures de données

Liste des structures de données avec une brève description :

Grid	5
Joueur	6
Origine	7





## Chapitre 2

# Index des fichiers

### 2.1 Liste des fichiers

Liste de tous les fichiers avec une brève description :

<a href="#">ajout_retrait_jetons.c</a>	Fichier source contenant les fonctions relatives à la gestion des jetons . . . . .	9
<a href="#">ajout_retrait_jetons.h</a>	Fichier header contenant les prototypes des fonctions relatives à la gestion des jetons . . . . .	11
<a href="#">check_winner.c</a>	Fichier source contenant les fonctions relatives au test d'un gagnant . . . . .	14
<a href="#">check_winner.h</a>	Fichier header contenant les prototypes des fonctions relatives au test d'un gagnant . . . . .	18
<a href="#">gestion_grille.c</a>	Fichier source contenant les fonctions relatives à la gestion de la grille . . . . .	24
<a href="#">gestion_grille.h</a>	Fichier header contenant les prototypes des fonctions relatives à la gestion de la grille . . . . .	27
<a href="#">header_principal.h</a>	Fichier header contenant les bibliothèques standards, les structures et les autres headers du projet . . . . .	31
<a href="#">jeu.c</a>	Fichier source contenant les fonctions relatives à la gestion du jeu solo/multi . . . . .	32
<a href="#">jeu.h</a>	Fichier header contenant les prototypes des fonctions relatives à la gestion du jeu solo/multi . . . . .	33
<a href="#">main.c</a>	Fichier source gérant l'intégralité du jeu Puissance N . . . . .	35
<a href="#">ordinateur.c</a>	Fichier source contenant les fonctions gérant les actions de l'IA de l'ordinateur . . . . .	36
<a href="#">ordinateur.h</a>	Fichier header contenant les prototypes des fonctions gérant les actions de l'IA de l'ordinateur . . . . .	37



## Chapitre 3

# Documentation des structures de données

### 3.1 Référence de la structure Grid

```
#include <header_principal.h>
```

#### Champs de données

- char \*\* grille
- int largeur
- int longueur

#### 3.1.1 Description détaillée

Structure définissant notre Grille : la grille (tableau 2D de caractères), sa largeur et sa longueur

Définition à la ligne 21 du fichier header\_principal.h.

#### 3.1.2 Documentation des champs

##### 3.1.2.1 grille

```
char** grille
```

Définition à la ligne 22 du fichier header\_principal.h.

### 3.1.2.2 largeur

```
int largeur
```

Définition à la ligne 23 du fichier header\_principal.h.

### 3.1.2.3 longueur

```
int longueur
```

Définition à la ligne 24 du fichier header\_principal.h.

La documentation de cette structure a été générée à partir du fichier suivant :

— [header\\_principal.h](#)

## 3.2 Référence de la structure Joueur

```
#include <header_principal.h>
```

### Champs de données

— char [couleur](#)  
— int [initiative](#)  
— char [jeton](#)

### 3.2.1 Description détaillée

Structure définissant les caractéristiques des joueurs : couleur, initiative et type de jeton

Définition à la ligne 38 du fichier header\_principal.h.

### 3.2.2 Documentation des champs

#### 3.2.2.1 couleur

```
char couleur
```

Définition à la ligne 39 du fichier header\_principal.h.

#### 3.2.2.2 initiative

```
int initiative
```

Définition à la ligne 40 du fichier header\_principal.h.

#### 3.2.2.3 jeton

```
char jeton
```

Définition à la ligne 41 du fichier header\_principal.h.

La documentation de cette structure a été générée à partir du fichier suivant :  
— [header\\_principal.h](#)

### 3.3 Référence de la structure Origine

```
#include <header_principal.h>
```

#### Champs de données

- int [abscisse](#)
- int [ordonnee](#)

#### 3.3.1 Description détaillée

Structure définissant les coordonnées du point d'origine des diagonales : abscisse et ordonnée

Définition à la ligne 30 du fichier header\_principal.h.

#### 3.3.2 Documentation des champs

##### 3.3.2.1 abscisse

```
int abscisse
```

Définition à la ligne 31 du fichier header\_principal.h.

##### 3.3.2.2 ordonnee

```
int ordonnee
```

Définition à la ligne 32 du fichier header\_principal.h.

La documentation de cette structure a été générée à partir du fichier suivant :  
— [header\\_principal.h](#)



# Chapitre 4

## Documentation des fichiers

### 4.1 Référence du fichier ajout\_retrait\_jetons.c

fichier source contenant les fonctions relatives à la gestion des jetons

```
#include "header_principal.h"
```

#### Fonctions

- int `colonne` (Grid Grille)
- int `add_token` (Grid \*Grille, int `colonne`, Joueur joueur)
- int `remove_token` (Grid \*Grille, int `colonne`, Joueur joueur, int nbjoueur)
- int `ligne_vide` (Grid Grille)
- int `grille_pleine` (Grid Grille)

#### 4.1.1 Description détaillée

fichier source contenant les fonctions relatives à la gestion des jetons

##### Auteur

Celiane A. & Loic B. & Mathis B.

##### Version

v1.0

##### Date

Juin 2021

#### 4.1.2 Documentation des fonctions

##### 4.1.2.1 `add_token()`

```
int add_token (  
    Grid * Grille,  
    int colonne,  
    Joueur joueur )
```

Permet d'ajouter, si possible, un jeton dans une colonne

**Paramètres**

<i>Grille</i>	: la grille de jeu et ses dimensions
<i>colonne</i>	: numéro de colonne où jouer
<i>joueur</i>	: les caractéristiques du joueur actif, couleur, initiative et type de jeton

**Renvoie**

1 si un jeton est correctement ajouté, 0 sinon

Définition à la ligne 39 du fichier ajout\_retrait\_jetons.c.

**4.1.2.2 colonne()**

```
int colonne (  
    Grid Grille )
```

Détermine le numéro de la colonne où le joueur souhaite effectuer une action

**Paramètres**

<i>Grille</i>	: la grille de jeu et ses dimensions
---------------	--------------------------------------

**Renvoie**

le numéro de la colonne où jouer

Définition à la ligne 16 du fichier ajout\_retrait\_jetons.c.

**4.1.2.3 grille\_pleine()**

```
int grille_pleine (  
    Grid Grille )
```

Permet de vérifier si la grille est pleine

**Paramètres**

<i>Grille</i>	: la grille de jeu et ses dimensions
---------------	--------------------------------------

**Renvoie**

1 si la grille est pleine, 0 sinon

Définition à la ligne 118 du fichier ajout\_retrait\_jetons.c.



#### 4.1.2.4 ligne\_vide()

```
int ligne_vide (
    Grid Grille )
```

Permet de verifier si la grille est vide

##### Paramètres

<i>Grille</i>	: la grille de jeu et ses dimensions
---------------	--------------------------------------

##### Renvoie

1 si la grille est vide, 0 sinon

Définition à la ligne 95 du fichier ajout\_retrait\_jetons.c.

#### 4.1.2.5 remove\_token()

```
int remove_token (
    Grid * Grille,
    int colonne,
    Joueur joueur,
    int nbjoueur )
```

Permet de retirer, si possible, un jeton dans une colonne

##### Paramètres

<i>Grille</i>	: la grille de jeu et ses dimensions
<i>colonne</i>	: numéro de colonne où jouer
<i>joueur</i>	: les caractéristiques du joueur actif, couleur, initiative et type de jeton

##### Renvoie

1 si un jeton est correctement retiré, 0 sinon

Définition à la ligne 67 du fichier ajout\_retrait\_jetons.c.

## 4.2 Référence du fichier ajout\_retrait\_jetons.h

fichier header contenant les prototypes des fonctions relatives à la gestion des jetons

```
#include "header_principal.h"
```

## Fonctions

- int `colonne` (`Grid` `Grille`)
- int `add_token` (`Grid` \*`Grille`, int `colonne`, `Joueur` `joueur`)
- int `remove_token` (`Grid` \*`Grille`, int `colonne`, `Joueur` `joueur`, int `nbjoueur`)
- int `ligne_vide` (`Grid` `Grille`)
- int `grille_pleine` (`Grid` `Grille`)

### 4.2.1 Description détaillée

fichier header contenant les prototypes des fonctions relatives à la gestion des jetons

#### Auteur

Celiane A. & Loic B. & Mathis B.

#### Version

v1.0

#### Date

Juin 2021

### 4.2.2 Documentation des fonctions

#### 4.2.2.1 `add_token()`

```
int add_token (  
    Grid * Grille,  
    int colonne,  
    Joueur joueur )
```

Permet d'ajouter, si possible, un jeton dans une colonne

#### Paramètres

<i>Grille</i>	: la grille de jeu et ses dimensions
<i>colonne</i>	: numéro de colonne où jouer
<i>joueur</i>	: les caractéristiques du joueur actif, couleur, initiative et type de jeton

#### Renvoie

1 si un jeton est correctement ajouté, 0 sinon

Définition à la ligne 39 du fichier `ajout_retrait_jetons.c`.

#### 4.2.2.2 colonne()

```
int colonne (  
    Grid Grille )
```

Détermine le numéro de la colonne où le joueur souhaite effectuer une action

##### Paramètres

<i>Grille</i>	: la grille de jeu et ses dimensions
---------------	--------------------------------------

##### Renvoie

le numéro de la colonne où jouer

Définition à la ligne 16 du fichier ajout\_retrait\_jetons.c.

#### 4.2.2.3 grille\_pleine()

```
int grille_pleine (  
    Grid Grille )
```

Permet de verifier si la grille est pleine

##### Paramètres

<i>Grille</i>	: la grille de jeu et ses dimensions
---------------	--------------------------------------

##### Renvoie

1 si la grille est pleine, 0 sinon

Définition à la ligne 118 du fichier ajout\_retrait\_jetons.c.

#### 4.2.2.4 ligne\_vide()

```
int ligne_vide (  
    Grid Grille )
```

Permet de verifier si la grille est vide

##### Paramètres

<i>Grille</i>	: la grille de jeu et ses dimensions
---------------	--------------------------------------

**Renvoie**

1 si la grille est vide, 0 sinon

Définition à la ligne 95 du fichier ajout\_retrait\_jetons.c.

**4.2.2.5 remove\_token()**

```
int remove_token (
    Grid * Grille,
    int colonne,
    Joueur joueur,
    int nbjoueur )
```

Permet de retirer, si possible, un jeton dans une colonne

**Paramètres**

<i>Grille</i>	: la grille de jeu et ses dimensions
<i>colonne</i>	: numéro de colonne où jouer
<i>joueur</i>	: les caractéristiques du joueur actif, couleur, initiative et type de jeton

**Renvoie**

1 si un jeton est correctement retiré, 0 sinon

Définition à la ligne 67 du fichier ajout\_retrait\_jetons.c.

**4.3 Référence du fichier check\_winner.c**

fichier source contenant les fonctions relatives au test d'un gagnant

```
#include "header_principal.h"
```

**Fonctions**

- int `check_winner` (Grid Grille, int colonne, Joueur joueur)
- int `check_verticale` (Grid Grille, int colonne, Joueur joueur)
- int `check_horizontale` (Grid Grille, int colonne, Joueur joueur)
- int `check_diagonale` (Grid Grille, int colonne, Joueur joueur)
- int `check_diagonale_croissante` (Grid Grille, int colonne, Joueur joueur)
- Origine `origine_diagonale_croissante` (Grid Grille, int colonne, Joueur joueur)
- int `check_diagonale_decroissante` (Grid Grille, int colonne, Joueur joueur)
- Origine `origine_diagonale_decroissante` (Grid Grille, int colonne, Joueur joueur)

### 4.3.1 Description détaillée

fichier source contenant les fonctions relatives au test d'un gagnant

#### Auteur

Celiane A. & Loic B. & Mathis B.

#### Version

v1.0

#### Date

Juin 2021

### 4.3.2 Documentation des fonctions

#### 4.3.2.1 check\_diagonale()

```
int check_diagonale (
    Grid Grille,
    int colonne,
    Joueur joueur )
```

Check si N jetons sont alignés en diagonale dans la grille

#### Paramètres

<i>Grille</i>	: la grille de jeu et ses dimensions
<i>colonne</i>	: numéro de colonne du dernier jeton ajouté
<i>joueur</i>	: caractéristiques du joueur venant de poser un jeton, couleur, initiative et type de jeton

#### Renvoie

le nombre de jeton alignés identiques consécutifs

Définition à la ligne 125 du fichier check\_winner.c.

#### 4.3.2.2 check\_diagonale\_croissante()

```
int check_diagonale_croissante (
    Grid Grille,
    int colonne,
    Joueur joueur )
```

Check si N jetons sont alignés en diagonale croissante (haut gauche vers bas droit) dans la grille

**Paramètres**

<i>Grille</i>	: la grille de jeu et ses dimensions
<i>colonne</i>	: numéro de colonne du dernier jeton ajouté
<i>joueur</i>	: caractéristiques du joueur venant de poser un jeton, couleur, initiative et type de jeton

**Renvoie**

le nombre de jeton alignés identiques consécutifs

Définition à la ligne 145 du fichier check\_winner.c.

**4.3.2.3 check\_diagonale\_decroissante()**

```
int check_diagonale_decroissante (  
    Grid Grille,  
    int colonne,  
    Joueur joueur )
```

Check si N jetons sont alignés en diagonale décroissante (bas gauche vers haut droit) dans la grille

**Paramètres**

<i>Grille</i>	: la grille de jeu et ses dimensions
<i>colonne</i>	: numéro de colonne du dernier jeton ajouté
<i>joueur</i>	: caractéristiques du joueur venant de poser un jeton, couleur, initiative et type de jeton

**Renvoie**

le nombre de jeton alignés identiques consécutifs

Définition à la ligne 216 du fichier check\_winner.c.

**4.3.2.4 check\_horizontal()**

```
int check_horizontal (  
    Grid Grille,  
    int colonne,  
    Joueur joueur )
```

Check si N jetons sont alignés horizontalement dans la grille

**Paramètres**

<i>Grille</i>	: la grille de jeu et ses dimensions
<i>colonne</i>	: numéro de colonne du dernier jeton ajouté
<i>joueur</i>	: caractéristiques du joueur venant de poser un jeton, couleur, initiative et type de jeton

**Renvoie**

le nombre de jeton alignés identiques consécutifs

Définition à la ligne 91 du fichier check\_winner.c.

**4.3.2.5 check\_verticale()**

```
int check_verticale (
    Grid Grille,
    int colonne,
    Joueur joueur )
```

Check si N jetons sont alignés verticalement dans la grille

**Paramètres**

<i>Grille</i>	: la grille de jeu et ses dimensions
<i>colonne</i>	: numéro de colonne du dernier jeton ajouté
<i>joueur</i>	: caractéristiques du joueur venant de poser un jeton, couleur, initiative et type de jeton

**Renvoie**

le nombre de jeton alignés identiques consécutifs

Définition à la ligne 63 du fichier check\_winner.c.

**4.3.2.6 check\_winner()**

```
int check_winner (
    Grid Grille,
    int colonne,
    Joueur joueur )
```

Check si N jetons sont alignés dans la grille

**Paramètres**

<i>Grille</i>	: la grille de jeu et ses dimensions
<i>colonne</i>	: numéro de colonne du dernier jeton ajouté
<i>joueur</i>	: caractéristiques du joueur venant de poser un jeton, couleur, initiative et type de jeton

**Renvoie**

1 si le joueur rouge gagne, 0 si c'est le joueur jaune, -1 sinon

Définition à la ligne 18 du fichier check\_winner.c.

#### 4.3.2.7 origine\_diagonale\_croissante()

```
Origine origine_diagonale_croissante (
    Grid Grille,
    int colonne,
    Joueur joueur )
```

Détermine le point à l'origine de la diagonale croissante dans la grille

##### Paramètres

<i>Grille</i>	: la grille de jeu et ses dimensions
<i>colonne</i>	: numéro de colonne du dernier jeton ajouté
<i>joueur</i>	: caractéristiques du joueur venant de poser un jeton, couleur, initiative et type de jeton

##### Renvoie

les coordonnées du point d'origine

Définition à la ligne 183 du fichier check\_winner.c.

#### 4.3.2.8 origine\_diagonale\_decroissante()

```
Origine origine_diagonale_decroissante (
    Grid Grille,
    int colonne,
    Joueur joueur )
```

Détermine le point à l'origine de la diagonale décroissante dans la grille

##### Paramètres

<i>Grille</i>	: la grille de jeu et ses dimensions
<i>colonne</i>	: numéro de colonne du dernier jeton ajouté
<i>joueur</i>	: caractéristiques du joueur venant de poser un jeton, couleur, initiative et type de jeton

##### Renvoie

les coordonnées du point d'origine

Définition à la ligne 253 du fichier check\_winner.c.

## 4.4 Référence du fichier check\_winner.h

fichier header contenant les prototypes des fonctions relatives au test d'un gagnant

```
#include "header_principal.h"
```



## Fonctions

- int `check_winner` (`Grid` Grille, int `colonne`, `Joueur` joueur)
- int `check_verticale` (`Grid` Grille, int `colonne`, `Joueur` joueur)
- int `check_horizontale` (`Grid` Grille, int `colonne`, `Joueur` joueur)
- int `check_diagonale` (`Grid` Grille, int `colonne`, `Joueur` joueur)
- int `check_diagonale_croissante` (`Grid` Grille, int `colonne`, `Joueur` joueur)
- `Origine` `origine_diagonale_croissante` (`Grid` Grille, int `colonne`, `Joueur` joueur)
- int `check_diagonale_decroissante` (`Grid` Grille, int `colonne`, `Joueur` joueur)
- `Origine` `origine_diagonale_decroissante` (`Grid` Grille, int `colonne`, `Joueur` joueur)

### 4.4.1 Description détaillée

fichier header contenant les prototypes des fonctions relatives au test d'un gagnant

#### Auteur

Celiane A. & Loic B. & Mathis B.

#### Version

v1.0

#### Date

Juin 2021

### 4.4.2 Documentation des fonctions

#### 4.4.2.1 `check_diagonale()`

```
int check_diagonale (  
    Grid Grille,  
    int colonne,  
    Joueur joueur )
```

Check si N jetons sont alignés en diagonale dans la grille

#### Paramètres

<i>Grille</i>	: la grille de jeu et ses dimensions
<i>colonne</i>	: numéro de colonne du dernier jeton ajouté
<i>joueur</i>	: caractéristiques du joueur venant de poser un jeton, couleur, initiative et type de jeton

#### Renvoie

le nombre de jeton alignés identiques consécutifs

Définition à la ligne 125 du fichier check\_winner.c.

#### 4.4.2.2 check\_diagonale\_croissante()

```
int check_diagonale_croissante (
    Grid Grille,
    int colonne,
    Joueur joueur )
```

Check si N jetons sont alignés en diagonale croissante (haut gauche vers bas droit) dans la grille

##### Paramètres

<i>Grille</i>	: la grille de jeu et ses dimensions
<i>colonne</i>	: numéro de colonne du dernier jeton ajouté
<i>joueur</i>	: caractéristiques du joueur venant de poser un jeton, couleur, initiative et type de jeton

##### Renvoie

le nombre de jeton alignés identiques consécutifs

Définition à la ligne 145 du fichier check\_winner.c.

#### 4.4.2.3 check\_diagonale\_decroissante()

```
int check_diagonale_decroissante (
    Grid Grille,
    int colonne,
    Joueur joueur )
```

Check si N jetons sont alignés en diagonale décroissante (bas gauche vers haut droit) dans la grille

##### Paramètres

<i>Grille</i>	: la grille de jeu et ses dimensions
<i>colonne</i>	: numéro de colonne du dernier jeton ajouté
<i>joueur</i>	: caractéristiques du joueur venant de poser un jeton, couleur, initiative et type de jeton

##### Renvoie

le nombre de jeton alignés identiques consécutifs

Définition à la ligne 216 du fichier check\_winner.c.

#### 4.4.2.4 check\_horizontal()

```
int check_horizontal (
    Grid Grille,
```

```
int colonne,  
Joueur joueur )
```

Check si N jetons sont alignés horizontalement dans la grille

**Paramètres**

<i>Grille</i>	: la grille de jeu et ses dimensions
<i>colonne</i>	: numéro de colonne du dernier jeton ajouté
<i>joueur</i>	: caractéristiques du joueur venant de poser un jeton, couleur, initiative et type de jeton

**Renvoie**

le nombre de jeton alignés identiques consécutifs

Définition à la ligne 91 du fichier check\_winner.c.

**4.4.2.5 check\_verticale()**

```
int check_verticale (
    Grid Grille,
    int colonne,
    Joueur joueur )
```

Check si N jetons sont alignés verticalement dans la grille

**Paramètres**

<i>Grille</i>	: la grille de jeu et ses dimensions
<i>colonne</i>	: numéro de colonne du dernier jeton ajouté
<i>joueur</i>	: caractéristiques du joueur venant de poser un jeton, couleur, initiative et type de jeton

**Renvoie**

le nombre de jeton alignés identiques consécutifs

Définition à la ligne 63 du fichier check\_winner.c.

**4.4.2.6 check\_winner()**

```
int check_winner (
    Grid Grille,
    int colonne,
    Joueur joueur )
```

Check si N jetons sont alignés dans la grille

**Paramètres**

<i>Grille</i>	: la grille de jeu et ses dimensions
<i>colonne</i>	: numéro de colonne du dernier jeton ajouté
<i>joueur</i>	: caractéristiques du joueur venant de poser un jeton, couleur, initiative et type de jeton

**Renvoie**

1 si le joueur rouge gagne, 0 si c'est le joueur jaune, -1 sinon

Définition à la ligne 18 du fichier check\_winner.c.

**4.4.2.7 origine\_diagonale\_croissante()**

```
Origine origine_diagonale_croissante (
    Grid Grille,
    int colonne,
    Joueur joueur )
```

Détermine le point à l'origine de la diagonale croissante dans la grille

**Paramètres**

<i>Grille</i>	: la grille de jeu et ses dimensions
<i>colonne</i>	: numéro de colonne du dernier jeton ajouté
<i>joueur</i>	: caractéristiques du joueur venant de poser un jeton, couleur, initiative et type de jeton

**Renvoie**

les coordonnées du point d'origine

Définition à la ligne 183 du fichier check\_winner.c.

**4.4.2.8 origine\_diagonale\_decroissante()**

```
Origine origine_diagonale_decroissante (
    Grid Grille,
    int colonne,
    Joueur joueur )
```

Détermine le point à l'origine de la diagonale décroissante dans la grille

**Paramètres**

<i>Grille</i>	: la grille de jeu et ses dimensions
<i>colonne</i>	: numéro de colonne du dernier jeton ajouté
<i>joueur</i>	: caractéristiques du joueur venant de poser un jeton, couleur, initiative et type de jeton

**Renvoie**

les coordonnées du point d'origine

Définition à la ligne 253 du fichier check\_winner.c.

## 4.5 Référence du fichier gestion\_grille.c

fichier source contenant les fonctions relatives à la gestion de la grille

```
#include "header_principal.h"
```

### Fonctions

- int `menu` ()
- void `initialisation` (Grid \*Grille)
- void `show_grid` (Grid Grille, Joueur joueur\_rouge)
- int `choix_nbjeueur` ()
- void `selection_joueur` (Joueur \*joueur\_jaune, Joueur \*joueur\_rouge, int nbjeueur)
- void `load` (Grid \*Grille, Joueur \*j1, Joueur \*j2, int \*nbjeueur)
- void `save` (Grid Grille, Joueur j1, Joueur j2, int nbjeueur)
- void `color` (int couleur\_texte, int couleur\_fond)
- void `free_memory` (Grid \*Grille)

### 4.5.1 Description détaillée

fichier source contenant les fonctions relatives à la gestion de la grille

#### Auteur

Celiane A. & Loic B. & Mathis B.

#### Version

v1.0

#### Date

Juin 2021

### 4.5.2 Documentation des fonctions

#### 4.5.2.1 choix\_nbjeueur()

```
int choix_nbjeueur ( )
```

Demande à l'utilisateur le nombre de joueurs, en cas d'une nouvelle partie

#### Renvoie

le nombre de joueurs

Définition à la ligne 180 du fichier gestion\_grille.c.

#### 4.5.2.2 color()

```
void color (
    int couleur_texte,
    int couleur_fond )
```

Permet de régler la couleur d'affichage des textes

## Paramètres

<i>couleur_texte</i>	: code (prédéfini) de la couleur du texte
<i>couleur_fond</i>	: code (prédéfini) de la couleur du fond

Définition à la ligne 392 du fichier gestion\_grille.c.

#### 4.5.2.3 free\_memory()

```
void free_memory (
    Grid * Grille )
```

Libère l'espace mémoire occupé par la grille en fin de programme

## Paramètres

<i>Grille</i>	: la grille de jeu et ses dimensions
---------------	--------------------------------------

Définition à la ligne 402 du fichier gestion\_grille.c.

#### 4.5.2.4 initialisation()

```
void initialisation (
    Grid * Grille )
```

Détermine la taille de la grille de jeu et initialise la grille avec des '\_'

## Paramètres

<i>Grille</i>	: la grille de jeu et ses dimensions
---------------	--------------------------------------

Définition à la ligne 63 du fichier gestion\_grille.c.

#### 4.5.2.5 load()

```
void load (
    Grid * Grille,
    Joueur * j1,
    Joueur * j2,
    int * nbjoueur )
```

Permet de charger une partie sauvegardée

**Paramètres**

<i>Grille</i>	: la grille et ses dimensions
<i>j1</i>	: les caractéristiques du premier joueur, couleur, initiative et type de jeton
<i>j2</i>	: les caractéristiques du deuxième joueur, couleur, initiative et type de jeton
<i>nbjoueur</i>	: nombre de joueurs (hors ordinateur)

Définition à la ligne 306 du fichier gestion\_grille.c.

**4.5.2.6 menu()**

```
int menu ( )
```

Propose au lancement du programme 3 options : nouvelle partie, lancer sauvegarde ou quitter

**Renvoie**

le numéro de l'action désirée

Définition à la ligne 15 du fichier gestion\_grille.c.

**4.5.2.7 save()**

```
void save (
    Grid Grille,
    Joueur j1,
    Joueur j2,
    int nbjoueur )
```

Permet de sauvegarder une partie

**Paramètres**

<i>Grille</i>	: la grille et ses dimensions
<i>j1</i>	: les caractéristiques du premier joueur, couleur, initiative et type de jeton
<i>j2</i>	: les caractéristiques du deuxième joueur, couleur, initiative et type de jeton
<i>nbjoueur</i>	: nombre de joueurs (hors ordinateur)

Définition à la ligne 356 du fichier gestion\_grille.c.

**4.5.2.8 selection\_joueur()**

```
void selection_joueur (
    Joueur * joueur_jaune,
```



```
Joueur * joueur_rouge,
int nbjoueur )
```

Initialise les caractéristiques des joueurs au lancement d'une partie

#### Paramètres

<i>joueur_jaune</i>	: les caractéristiques du joueur jaune, couleur, initiative et type de jeton
<i>joueur_rouge</i>	: les caractéristiques du joueur rouge, couleur, initiative et type de jeton
<i>nbjoueur</i>	: nombre de joueurs (hors ordinateur)

Définition à la ligne 205 du fichier gestion\_grille.c.

#### 4.5.2.9 show\_grid()

```
void show_grid (
    Grid Grille,
    Joueur joueur_rouge )
```

Permet d'afficher la grille de jeu

#### Paramètres

<i>Grille</i>	: la grille de jeu et ses dimensions
<i>joueur_rouge</i>	: les caractéristiques du joueur rouge, couleur, initiative et type de jeton

Définition à la ligne 102 du fichier gestion\_grille.c.

## 4.6 Référence du fichier gestion\_grille.h

fichier header contenant les prototypes des fonctions relatives à la gestion de la grille

```
#include "header_principal.h"
```

### Fonctions

- void **initialisation** (Grid \*Grille)
- void **show\_grid** (Grid Grille, Joueur joueur\_rouge)
- int **menu** ()
- int **choix\_nbjoueur** ()
- void **selection\_joueur** (Joueur \*joueur\_jaune, Joueur \*joueur\_rouge, int nbjoueur)
- void **color** (int couleur\_texte, int couleur\_fond)
- void **save** (Grid Grille, Joueur j1, Joueur j2, int nbjoueur)
- void **load** (Grid \*Grille, Joueur \*j1, Joueur \*j2, int \*nbjoueur)
- void **free\_memory** (Grid \*Grille)

### 4.6.1 Description détaillée

fichier header contenant les prototypes des fonctions relatives à la gestion de la grille

#### Auteur

Celiane A. & Loic B. & Mathis B.

#### Version

v1.0

#### Date

Juin 2021

### 4.6.2 Documentation des fonctions

#### 4.6.2.1 choix\_nbjeueur()

```
int choix_nbjeueur ( )
```

Demande à l'utilisateur le nombre de joueurs, en cas d'une nouvelle partie

#### Renvoie

le nombre de joueurs

Définition à la ligne 180 du fichier gestion\_grille.c.

#### 4.6.2.2 color()

```
void color (
    int couleur_texte,
    int couleur_fond )
```

Permet de régler la couleur d'affichage des textes

#### Paramètres

<i>couleur_texte</i>	: code (prédéfini) de la couleur du texte
<i>couleur_fond</i>	: code (prédéfini) de la couleur du fond

Définition à la ligne 392 du fichier gestion\_grille.c.

#### 4.6.2.3 free\_memory()

```
void free_memory (
    Grid * Grille )
```

Libère l'espace mémoire occupé par la grille en fin de programme

##### Paramètres

<i>Grille</i>	: la grille de jeu et ses dimensions
---------------	--------------------------------------

Définition à la ligne 402 du fichier gestion\_grille.c.

#### 4.6.2.4 initialisation()

```
void initialisation (
    Grid * Grille )
```

Détermine la taille de la grille de jeu et initialise la grille avec des '\_'

##### Paramètres

<i>Grille</i>	: la grille de jeu et ses dimensions
---------------	--------------------------------------

Détermine la taille de la grille de jeu et initialise la grille avec des '\_'

##### Paramètres

<i>Grille</i>	: la grille de jeu et ses dimensions
---------------	--------------------------------------

Définition à la ligne 63 du fichier gestion\_grille.c.

#### 4.6.2.5 load()

```
void load (
    Grid * Grille,
    Joueur * j1,
    Joueur * j2,
    int * nbjoueur )
```

Permet de charger une partie sauvegardée

##### Paramètres

<i>Grille</i>	: la grille et ses dimensions
<i>j1</i>	: les caractéristiques du premier joueur, couleur, initiative et type de jeton
<i>j2</i>	: les caractéristiques du deuxième joueur, couleur, initiative et type de jeton
<i>nbjoueur</i>	: nombre de joueurs (hors ordinateur)

Définition à la ligne 306 du fichier gestion\_grille.c.

#### 4.6.2.6 menu()

```
int menu ( )
```

Propose au lancement du programme 3 options : nouvelle partie, lancer sauvegarde ou quitter

##### Renvoie

le numéro de l'action désirée

Définition à la ligne 15 du fichier gestion\_grille.c.

#### 4.6.2.7 save()

```
void save (
    Grid Grille,
    Joueur j1,
    Joueur j2,
    int nbjoueur )
```

Permet de sauvegarder une partie

##### Paramètres

<i>Grille</i>	: la grille et ses dimensions
<i>j1</i>	: les caractéristiques du premier joueur, couleur, initiative et type de jeton
<i>j2</i>	: les caractéristiques du deuxième joueur, couleur, initiative et type de jeton
<i>nbjoueur</i>	: nombre de joueurs (hors ordinateur)

Définition à la ligne 356 du fichier gestion\_grille.c.

#### 4.6.2.8 selection\_joueur()

```
void selection_joueur (
    Joueur * joueur_jaune,
    Joueur * joueur_rouge,
    int nbjoueur )
```

Initialise les caractéristiques des joueurs au lancement d'une partie

## Paramètres

<i>joueur_jaune</i>	: les caractéristiques du joueur jaune, couleur, initiative et type de jeton
<i>joueur_rouge</i>	: les caractéristiques du joueur rouge, couleur, initiative et type de jeton
<i>nbjoueur</i>	: nombre de joueurs (hors ordinateur)

Définition à la ligne 205 du fichier gestion\_grille.c.

## 4.6.2.9 show\_grid()

```
void show_grid (
    Grid Grille,
    Joueur joueur_rouge )
```

Permet d'afficher la grille de jeu

## Paramètres

<i>Grille</i>	: la grille de jeu et ses dimensions
<i>joueur_rouge</i>	: les caractéristiques du joueur rouge, couleur, initiative et type de jeton

Définition à la ligne 102 du fichier gestion\_grille.c.

## 4.7 Référence du fichier header\_principal.h

fichier header contenant les bibliothèques standards, les structures et les autres headers du projet

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#include <windows.h>
#include <conio.h>
#include "gestion_grille.h"
#include "ajout_retrait_jetons.h"
#include "jeu.h"
#include "ordinateur.h"
#include "check_winner.h"
```

## Structures de données

- struct [Grid](#)
- struct [Origine](#)
- struct [Joueur](#)

### 4.7.1 Description détaillée

fichier header contenant les bibliothèques standards, les structures et les autres headers du projet

**Auteur**

Celiane A. & Loic B. & Mathis B.

**Version**

v1.0

**Date**

Juin 2021

## 4.8 Référence du fichier jeu.c

fichier source contenant les fonctions relatives à la gestion du jeu solo/multi

```
#include "header_principal.h"
```

### Fonctions

- void [affichage\\_trophee](#) ()
- void [jeu\\_multi](#) ([Grid](#) Grille, [Joueur](#) joueur\_rouge, [Joueur](#) joueur\_jaune)
- void [jeu\\_solo](#) ([Grid](#) Grille, [Joueur](#) joueur\_rouge, [Joueur](#) joueur\_jaune)

### 4.8.1 Description détaillée

fichier source contenant les fonctions relatives à la gestion du jeu solo/multi

**Auteur**

Celiane A. & Loic B. & Mathis B.

**Version**

v1.0

**Date**

Juin 2021

### 4.8.2 Documentation des fonctions

#### 4.8.2.1 affichage\_trophee()

```
void affichage_trophee ( )
```

Permet l'affichage d'un trophée en cas de victoire d'un joueur réel

Définition à la ligne 14 du fichier jeu.c.

#### 4.8.2.2 jeu\_multi()

```
void jeu_multi (
    Grid Grille,
    Joueur joueur_rouge,
    Joueur joueur_jaune )
```

Assure le déroulement du jeu à 2 joueurs

##### Paramètres

<i>Grille</i>	: la grille de jeu et ses dimensions
<i>joueur_rouge</i>	: les caractéristiques du joueur rouge, couleur, initiative et type de jeton
<i>joueur_jaune</i>	: les caractéristiques du joueur jaune, couleur, initiative et type de jeton

Définition à la ligne 50 du fichier jeu.c.

#### 4.8.2.3 jeu\_solo()

```
void jeu_solo (
    Grid Grille,
    Joueur joueur_rouge,
    Joueur joueur_jaune )
```

Assure le déroulement du jeu joueur vs ordinateur

##### Paramètres

<i>Grille</i>	: la Grille de jeu et ses dimensions
<i>joueur_rouge</i>	: les caractéristiques du joueur rouge, couleur, initiative et type de jeton
<i>joueur_jaune</i>	: les caractéristiques du joueur "ordinateur", couleur, initiative et type de jeton

Définition à la ligne 164 du fichier jeu.c.

## 4.9 Référence du fichier jeu.h

fichier header contenant les prototypes des fonctions relatives à la gestion du jeu solo/multi

```
#include "header_principal.h"
```

## Fonctions

- void `affichage_trophee` ()
- void `jeu_multi` (`Grid` Grille, `Joueur` joueur\_rouge, `Joueur` joueur\_jaune)
- void `jeu_solo` (`Grid` Grille, `Joueur` joueur\_rouge, `Joueur` joueur\_jaune)

### 4.9.1 Description détaillée

fichier header contenant les prototypes des fonctions relatives à la gestion du jeu solo/multi

#### Auteur

Celiane A. & Loic B. & Mathis B.

#### Version

v1.0

#### Date

Juin 2021

### 4.9.2 Documentation des fonctions

#### 4.9.2.1 `affichage_trophee()`

```
void affichage_trophee ( )
```

Permet l'affichage d'un trophée en cas de victoire d'un joueur réel

Définition à la ligne 14 du fichier jeu.c.

#### 4.9.2.2 `jeu_multi()`

```
void jeu_multi (
    Grid Grille,
    Joueur joueur_rouge,
    Joueur joueur_jaune )
```

Assure le déroulement du jeu à 2 joueurs



## Paramètres

<i>Grille</i>	: la grille de jeu et ses dimensions
<i>joueur_rouge</i>	: les caractéristiques du joueur rouge, couleur, initiative et type de jeton
<i>joueur_jaune</i>	: les caractéristiques du joueur jaune, couleur, initiative et type de jeton

Définition à la ligne 50 du fichier jeu.c.

## 4.9.2.3 jeu\_solo()

```
void jeu_solo (
    Grid Grille,
    Joueur joueur_rouge,
    Joueur joueur_jaune )
```

Assure le déroulement du jeu joueur vs ordinateur

## Paramètres

<i>Grille</i>	: la Grille de jeu et ses dimensions
<i>joueur_rouge</i>	: les caractéristiques du joueur rouge, couleur, initiative et type de jeton
<i>joueur_jaune</i>	: les caractéristiques du joueur "ordinateur", couleur, initiative et type de jeton

Définition à la ligne 164 du fichier jeu.c.

## 4.10 Référence du fichier main.c

fichier source gérant l'intégralité du jeu Puissance N

```
#include "header_principal.h"
```

## Fonctions

— int `main` ()

## 4.10.1 Description détaillée

fichier source gérant l'intégralité du jeu Puissance N

## Auteur

Celiane A. & Loic B. & Mathis B.

## Version

v1.0

## Date

Juin 2021

## 4.10.2 Documentation des fonctions

### 4.10.2.1 main()

```
int main ( )
```

Définition à la ligne 11 du fichier main.c.

## 4.11 Référence du fichier ordinateur.c

fichier source contenant les fonctions gérant les actions de l'IA de l'ordinateur

```
#include "header_principal.h"
```

### Fonctions

- int `colonne_ordi` ([Grid](#) Grille)
- int `choix_action_ordi` ([Grid](#) Grille)

### 4.11.1 Description détaillée

fichier source contenant les fonctions gérant les actions de l'IA de l'ordinateur

#### Auteur

Celiane A. & Loic B. & Mathis B.

#### Version

v1.0

#### Date

Juin 2021

## 4.11.2 Documentation des fonctions

### 4.11.2.1 choix\_action\_ordi()

```
int choix_action_ordi (
    Grid Grille )
```

Détermine le choix de l'ordinateur : ajouter ou retirer un jeton

## Paramètres

<i>Grille</i>	: la grille de jeu et ses dimensions
---------------	--------------------------------------

## Renvoie

le choix de l'ordinateur

Définition à la ligne 32 du fichier ordinateur.c.

## 4.11.2.2 colonne\_orði()

```
int colonne_orði (
    Grid Grille )
```

Détermine le numéro de la colonne où l'ordinateur souhaite effectuer une action

## Paramètres

<i>Grille</i>	: la grille de jeu et ses dimensions
---------------	--------------------------------------

## Renvoie

le numéro de la colonne où jouer

Définition à la ligne 16 du fichier ordinateur.c.

## 4.12 Référence du fichier ordinateur.h

fichier header contenant les prototypes des fonctions gérant les actions de l'IA de l'ordinateur

```
#include "header_principal.h"
```

## Fonctions

- int colonne\_orði (Grid Grille)
- int choix\_action\_orði (Grid Grille)

## 4.12.1 Description détaillée

fichier header contenant les prototypes des fonctions gérant les actions de l'IA de l'ordinateur

## Auteur

Celiane A. & Loic B. & Mathis B.

## Version

v1.0

## Date

Juin 2021

## 4.12.2 Documentation des fonctions

### 4.12.2.1 choix\_action\_ordi()

```
int choix_action_ordi (
    Grid Grille )
```

Détermine le choix de l'ordinateur : ajouter ou retirer un jeton

#### Paramètres

<i>Grille</i>	: la grille de jeu et ses dimensions
---------------	--------------------------------------

#### Renvoie

le choix de l'ordinateur

Définition à la ligne 32 du fichier ordinateur.c.

### 4.12.2.2 colonne\_ordi()

```
int colonne_ordi (
    Grid Grille )
```

Détermine le numéro de la colonne où l'ordinateur souhaite effectuer une action

#### Paramètres

<i>Grille</i>	: la grille de jeu et ses dimensions
---------------	--------------------------------------

#### Renvoie

le numéro de la colonne où jouer

Définition à la ligne 16 du fichier ordinateur.c.

# Index

- abscisse
  - Origine, [7](#)
- add\_token
  - ajout\_retrait\_jetons.c, [9](#)
  - ajout\_retrait\_jetons.h, [12](#)
- affichage\_trophee
  - jeu.c, [32](#)
  - jeu.h, [34](#)
- ajout\_retrait\_jetons.c, [9](#)
  - add\_token, [9](#)
  - colonne, [10](#)
  - grille\_pleine, [10](#)
  - ligne\_vide, [10](#)
  - remove\_token, [11](#)
- ajout\_retrait\_jetons.h, [11](#)
  - add\_token, [12](#)
  - colonne, [12](#)
  - grille\_pleine, [13](#)
  - ligne\_vide, [13](#)
  - remove\_token, [14](#)
- check\_diagonale
  - check\_winner.c, [15](#)
  - check\_winner.h, [19](#)
- check\_diagonale\_croissante
  - check\_winner.c, [15](#)
  - check\_winner.h, [19](#)
- check\_diagonale\_decroissante
  - check\_winner.c, [16](#)
  - check\_winner.h, [20](#)
- check\_horizontal
  - check\_winner.c, [16](#)
  - check\_winner.h, [20](#)
- check\_verticale
  - check\_winner.c, [17](#)
  - check\_winner.h, [22](#)
- check\_winner
  - check\_winner.c, [17](#)
  - check\_winner.h, [22](#)
- check\_winner.c, [14](#)
  - check\_diagonale, [15](#)
  - check\_diagonale\_croissante, [15](#)
  - check\_diagonale\_decroissante, [16](#)
  - check\_horizontal, [16](#)
  - check\_verticale, [17](#)
  - check\_winner, [17](#)
  - origine\_diagonale\_croissante, [17](#)
  - origine\_diagonale\_decroissante, [18](#)
- check\_winner.h, [18](#)
  - check\_diagonale, [19](#)
- check\_diagonale\_croissante, [19](#)
- check\_diagonale\_decroissante, [20](#)
- check\_horizontal, [20](#)
- check\_verticale, [22](#)
- check\_winner, [22](#)
- origine\_diagonale\_croissante, [23](#)
- origine\_diagonale\_decroissante, [23](#)
- choix\_action\_ori
  - ordinateur.c, [36](#)
  - ordinateur.h, [38](#)
- choix\_nbjeueur
  - gestion\_grille.c, [24](#)
  - gestion\_grille.h, [28](#)
- colonne
  - ajout\_retrait\_jetons.c, [10](#)
  - ajout\_retrait\_jetons.h, [12](#)
- colonne\_ori
  - ordinateur.c, [37](#)
  - ordinateur.h, [38](#)
- color
  - gestion\_grille.c, [24](#)
  - gestion\_grille.h, [28](#)
- couleur
  - Joueur, [6](#)
- free\_memory
  - gestion\_grille.c, [25](#)
  - gestion\_grille.h, [28](#)
- gestion\_grille.c, [24](#)
  - choix\_nbjeueur, [24](#)
  - color, [24](#)
  - free\_memory, [25](#)
  - initialisation, [25](#)
  - load, [25](#)
  - menu, [26](#)
  - save, [26](#)
  - selection\_joueur, [26](#)
  - show\_grid, [27](#)
- gestion\_grille.h, [27](#)
  - choix\_nbjeueur, [28](#)
  - color, [28](#)
  - free\_memory, [28](#)
  - initialisation, [29](#)
  - load, [29](#)
  - menu, [30](#)
  - save, [30](#)
  - selection\_joueur, [30](#)
  - show\_grid, [31](#)
- Grid, [5](#)

- grille, [5](#)
- largeur, [5](#)
- longueur, [6](#)
- grille
  - Grid, [5](#)
- grille\_pleine
  - ajout\_retrait\_jetons.c, [10](#)
  - ajout\_retrait\_jetons.h, [13](#)
- header\_principal.h, [31](#)
- initialisation
  - gestion\_grille.c, [25](#)
  - gestion\_grille.h, [29](#)
- initiative
  - Joueur, [6](#)
- jeton
  - Joueur, [7](#)
- jeu.c, [32](#)
  - affichage\_trophee, [32](#)
  - jeu\_multi, [33](#)
  - jeu\_solo, [33](#)
- jeu.h, [33](#)
  - affichage\_trophee, [34](#)
  - jeu\_multi, [34](#)
  - jeu\_solo, [35](#)
- jeu\_multi
  - jeu.c, [33](#)
  - jeu.h, [34](#)
- jeu\_solo
  - jeu.c, [33](#)
  - jeu.h, [35](#)
- Joueur, [6](#)
  - couleur, [6](#)
  - initiative, [6](#)
  - jeton, [7](#)
- largeur
  - Grid, [5](#)
- ligne\_vide
  - ajout\_retrait\_jetons.c, [10](#)
  - ajout\_retrait\_jetons.h, [13](#)
- load
  - gestion\_grille.c, [25](#)
  - gestion\_grille.h, [29](#)
- longueur
  - Grid, [6](#)
- main
  - main.c, [36](#)
- main.c, [35](#)
  - main, [36](#)
- menu
  - gestion\_grille.c, [26](#)
  - gestion\_grille.h, [30](#)
- ordinateur.c, [36](#)
  - choix\_action\_ordi, [36](#)
  - colonne\_ordi, [37](#)
- ordinateur.h, [37](#)
  - choix\_action\_ordi, [38](#)
  - colonne\_ordi, [38](#)
- ordonnee
  - Origine, [7](#)
- Origine, [7](#)
  - abscisse, [7](#)
  - ordonnee, [7](#)
- origine\_diagonale\_croissante
  - check\_winner.c, [17](#)
  - check\_winner.h, [23](#)
- origine\_diagonale\_decroissante
  - check\_winner.c, [18](#)
  - check\_winner.h, [23](#)
- remove\_token
  - ajout\_retrait\_jetons.c, [11](#)
  - ajout\_retrait\_jetons.h, [14](#)
- save
  - gestion\_grille.c, [26](#)
  - gestion\_grille.h, [30](#)
- selection\_joueur
  - gestion\_grille.c, [26](#)
  - gestion\_grille.h, [30](#)
- show\_grid
  - gestion\_grille.c, [27](#)
  - gestion\_grille.h, [31](#)