三目並べ 詳細仕様書

Cチーム  
メンバー  
村上 海斗　川島 冬嗣

内容

[1. 概要 1](#_Toc166662571)

[2. 語句定義 1](#_Toc166662572)

[3. ファイル説明 1](#_Toc166662573)

[3.1 ファイル構成 1](#_Toc166662574)

[3.2 ヘッダファイル 2](#_Toc166662575)

[4. 関数説明 4](#_Toc166662576)

[4.1 関数一覧 4](#_Toc166662577)

[4.2 main関数 6](#_Toc166662578)

[5. 機能構成 7](#_Toc166662579)

[5.1 機能一覧 7](#_Toc166662580)

[5.2 フローチャート 9](#_Toc166662581)

[6. 画面構成 11](#_Toc166662582)

# 概要

本書は、「三目並べ」のファイル構成、関数説明、機能構成、画面構成についてまとめた詳細仕様書です。

表 1語句定義

# 語句定義

本書で記載される語句は、以下の通り定義します。

|  |  |  |
| --- | --- | --- |
| No | 語句 | 定義 |
| 1 | 勝利条件 | 駒が縦、横、斜めどれか一列揃った場合 |
| 2 | 引き分け条件 | 勝利条件を満たさず、盤面が駒で埋まった場合 |
| 3 | 終了条件 | 勝利条件、引き分け条件を満たした場合 |
| 4 | 再プレイ | もう一度、三目並べで遊ぶこと |

# ファイル説明

## ファイル構成

ファイル構成は図 1の通りです。各ファイルの役割を表 2に示しています。

グラフ, ウォーターフォール図

自動的に生成された説明

図 1ファイル構成

表 2各ファイルの役割

|  |  |
| --- | --- |
| ファイル名 | 役割 |
| main.c/main.h | main関数、三目並べのゲーム進行を制御する。 |
| tutorial.h / tutorial.c | チュートリアルを表示する。 |
| board.h / board.c | 三目並べの盤面を制御する。 |

## ヘッダファイル

各ヘッダファイルで使用する定義を表 3～表 5に示しています。

表 3main.h

|  |  |  |
| --- | --- | --- |
| 定義名・宣言名 | 定義・宣言 | 説明 |
| マクロ | NAME\_LEN (10) | プレイヤー名の長さ上限 |
| FALSE (0) | “FALSE”を0とする |
| TRUE (1) | “TRUE”を1とする |
| enum型 | TURN | FIRST\_TURN（先手のターン）、SECOND\_TURN（後手のターン）を定義 |
| 構造体 | typedef struct PLAYER | 構造体の中身は、プレイヤー名を表す name (char\*型)、 駒の種類を表す piece(char型)とする。 |

表 4board.h

|  |  |  |
| --- | --- | --- |
| 定義名・宣言名 | 定義・宣言 | 説明 |
| マクロ | BOARD\_SIZE (3) | マスのサイズを3×3の設定 |
| enum型 | RESULT | FIRST\_WIN（先手勝利）、SECOND\_WIN（後手勝利）、DROW（引き分け）、NONE（未確定） の4つを定義 |
| プロトタイプ | void print\_now\_board(void) | プロトタイプ宣言 |
| int put\_piece(int row, int column PLAYER player) |
| RESULT judge(void) |

表 5tutorial.h

|  |  |  |
| --- | --- | --- |
| 定義名・宣言名 | 定義・宣言 | 説明 |
| プロトタイプ | void view\_tutorial(void) | プロトタイプ宣言 |

# 関数説明

## 関数一覧

使用する関数は、以下の表 6の通りです。

表 6関数の説明

|  |  |  |
| --- | --- | --- |
| ファイル名 | 関数名 | 説明 |
| tutorial.c | void view\_tutorial(void) | main関数で呼び出される。チュートリアルを表示する。 |
| board.c | void print\_now\_board(void) | game\_progress関数内で呼び出される。現在の盤面を表示する。 |
| int put\_piece(int row, int column, Player player) | game\_progress関数内で呼び出される。row(行)column(列)の位置に、playerのpieceを配置する。駒を配置出来た場合にはTRUE(1)、配置出来なかった場合にはFALSE(0)を戻り値として返す。 |
| RESULT judge(void) | game\_progress関数内で呼び出される。終了条件に当てはまるか判断する。引き分けならDROW、先手が勝ちならFIRST\_WIN、後手が勝ちならSECOND\_WIN、未確定ならNONEを戻り値として返す。 |
| main.c | TURN change\_turn(TURN now) | game\_progress関数内で呼び出される。nowが先手なら後手を、nowが後手なら先手を戻り値として返す。 |
| void game\_progress(char\* name1, char\* name2) | main関数内で呼び出される。ゲームの進行を行う。「name1、’O’」と「name2、’X’」の組み合わせで、それぞれPLAYER型変数に格納する。 |
| void print\_result(RESULT result) | game\_progress関数内で呼び出される。resultで受け取った値に応じてゲーム結果を表示する。resultにはenum定義したRESULTのいずれかが入る。 |
| int main(void) | メイン関数は4.3で詳しく説明。 |

## main関数

main関数での処理の手順について説明します。

1. ユーザからの入力を受け、入力が半角数字1であればview\_tutorial関数を呼び出す。1以外の入力であれば次の処理へ移る。
2. 先手、後手のプレイヤー名の入力を受け、main関数内で作成する任意の変数2つにそれぞれ格納する。
3. game\_progress関数を呼び出す。引数には、②でプレイヤー名を格納した変数を入れる。
4. retry\_game関数を呼び出し、戻り値がTRUE(1)なら③の処理に戻る。FALSE(0)であれば次の処理へ移る。
5. “ゲーム終了”を表示し、終了する。

# 機能構成

## 機能一覧

入力機能、出力機能の一覧と説明を表 7、表 8に示しています。

表 7入力機能一覧

|  |  |  |
| --- | --- | --- |
| No. | 入力機能一覧 | 説明 |
| 01 | チュートリアル表示確認 | main関数内で、ユーザ入力を受け、半角数字1が入力されるとview\_tutorial関数を呼び出す。1以外の入力の場合、処理は行わない。 |
| 02 | プレイヤー名入力 | main関数内で、ユーザ入力を受け、プレイヤー名を決定する。プレイヤー名は半角英数字10字以下としており、それを満たさない入力の場合、警告文を表示し、再度入力要求する。 |
| 03 | 座標入力 | game\_progress関数内でユーザ入力を受け、put\_piece関数を呼び出し、入力された座標に駒を配置する。入力方法は、〇行×列を「〇␣×」(〇と×の間は半角スペース)の形で入力する。座標が不正な場合（既に駒が配置済み、マスのサイズよりも大きい）、警告文を表示し、再度入力要求する。 |
| 04 | 再プレイの有無確認 | main関数内retry\_game関数を呼び出し、半角数字1が入力されると、再度三目並べを行う。1以外の入力の場合、処理は行わない。 |

表 8出力機能一覧

|  |  |  |
| --- | --- | --- |
| No. | 出力機能一覧 | 説明 |
| 01 | チュートリアル表示 | main関数内でview\_tutorial関数を呼び出しチュートリアルを表示する。 |
| 02 | 盤面表示 | game\_progress関数内でprint\_now\_board関数を呼び出し、現在の盤面を表示する。行と列を見やすくし、先手が配置した駒を’O’、後手が配置した駒を’X’と表示する。 |
| 03 | 結果表示 | game\_progress関数内でprint\_result関数を呼び出し、結果を表示する。勝利条件を満たした場合、勝者を表示、引き分け条件を満たした場合、引き分けを表示する。 |

## フローチャート

フローチャートを図 2に示しています。

ダイアグラム

自動的に生成された説明

図 3フローチャート

図 4機能8

# 画面構成

画面構成を図 5、図 6に示します。これらの図はチュートリアルに表示する内容に、緑字で説明を加えたものです。そのため、実装の際には緑字で書かれた文章は出力しないものとします。また、青字で書かれた文章はチュートリアル内のみでの出力とします。

テキスト

自動的に生成された説明

図 5画面構成1

テキスト

中程度の精度で自動的に生成された説明

図 6画面構成2