三目並べ　詳細仕様書（改訂版1）

チームJ 岩田碧海琉　中川誠淳

目次

[1. 概要 2](#_Toc167104292)

[2. ファイル構成 2](#_Toc167104293)

[2.1. ソースコードファイル 2](#_Toc167104294)

[2.3. 関数 2](#_Toc167104295)

[2.3. 定義する列挙型および構造体 3](#_Toc167104296)

[2.3.1 構造体 3](#_Toc167104297)

[2.3.2 列挙型 3](#_Toc167104298)

[3. 機能構成 3](#_Toc167104299)

[3.1. ｍain.cファイルの機能 3](#_Toc167104300)

[3.1.1. 配列の更新 3](#_Toc167104301)

[3.1.2. カウント 3](#_Toc167104302)

[3.1.3. プレイヤーの交代 3](#_Toc167104303)

[3.1.4. 勝敗の表示 3](#_Toc167104304)

[3.1.5. リトライ機能 4](#_Toc167104305)

[3.2. print\_grid.cファイルの機能 6](#_Toc167104306)

[3.3. player\_input.cファイルの機能 6](#_Toc167104307)

[3.3.1. 入力受け取り機能 6](#_Toc167104308)

[3.3.2. 入力受け取りエラー機能 7](#_Toc167104309)

[3.4. game\_judge.cファイルの機能 9](#_Toc167104310)

[4. 画面構成 11](#_Toc167104311)

# 概要

本仕様書は、ファイル構成、機能構成、画面構成、これら３つの構成を順に示す。

# ファイル構成

* 1. ソースコードファイル

表1：各ファイル名と内容

|  |  |
| --- | --- |
| ファイル名 | ファイルの内容 |
| main.c | プログラムのメインファイル。ゲームの進行と各機能の統合を担当 |
| print\_grid.c | 盤面の表示を担当。Print\_Grid()関数 |
| player\_input.c | プレイヤーの入力を担当。Player\_Input()関数 |
| game\_judge.c | ゲームの勝敗の判定を担当。Game\_Judge()関数 |

* 1. 本仕様書にて定義する関数

Print\_Grid関数

引数：盤面の情報を格納する配列（board）のchar型先頭ポインタ

返り値：なし

Player\_Input関数

引数：盤面の情報を格納する配列（board）のchar型先頭ポインタ

返り値：プレイヤーが指定したマスの座標(x,y)を格納する構造体BOARD\_XY

Game\_Judge関数

引数：ターンのカウント(int型)、プレイヤーが指定したマスの座標(x,y)を格納する構造体BOARD\_XY、盤面の情報を格納する配列（board）のchar型先頭ポインタ

返り値：勝ち、引き分け、ゲーム継続の情報をもつ列挙型RESULT

* 1. 本仕様書にて定義する列挙型、構造体、配列および変数
     + 1. 構造体

・構造体タグ

BOARD\_XY

・メンバ

int型 x：盤面のx座標

int型 y：盤面のｙ座標

* + 1. 列挙型

・列挙体タグ

RESULT

・列挙定数

｛NONE,WIN, DRAW｝

NONE：勝敗がつかず、かつ引き分けでもない状態

WIN：player1またはplayer2が勝利

DRAW：引き分け

1. 配列

盤面の情報を格納する配列「board」

・char型

・３行３列の二次元配列

・半角スペース’ ’で初期化

1. 変数

ターンのカウント

・int型

ただし、本仕様書で定義したもの以外にも適宜変数などを定義してよい

# 機能構成

* 1. ｍain.cファイルの機能
     1. 配列の更新

配列の初期値は、半角スぺース’ ‘とし、プレイヤーの座標入力に応じて、配列の要素を指定の記号で更新する。player1ならば半角文字オー‘o’で、player2ならば半角文字エックス’x’で置き換える。

* + 1. カウント

マスへ入力する回数（ターン）をカウントする。カウントの初期値は１とする。

* + 1. プレイヤーの交代

カウントをもとに、入力プレイヤーを交代する。交代時、および初手の時、手番のプレイヤー名を「player1のターン」のように表示する(player2のときも同様)。カウントが奇数の時player1、偶数の時player2となる。カウントの初期値は１とする。また、初手プレイヤーはplayer1とする。

* + 1. 勝敗の表示

Game\_Judge()から判定結果を返り値として受け取り、

勝利時には

と勝利プレイヤーを表示する。（player2も同様に表示）

ノートパソコン, コンピュータ, 猫, 画面 が含まれている画像

自動的に生成された説明引き分け時には

と表示する。

* + 1. リトライ機能

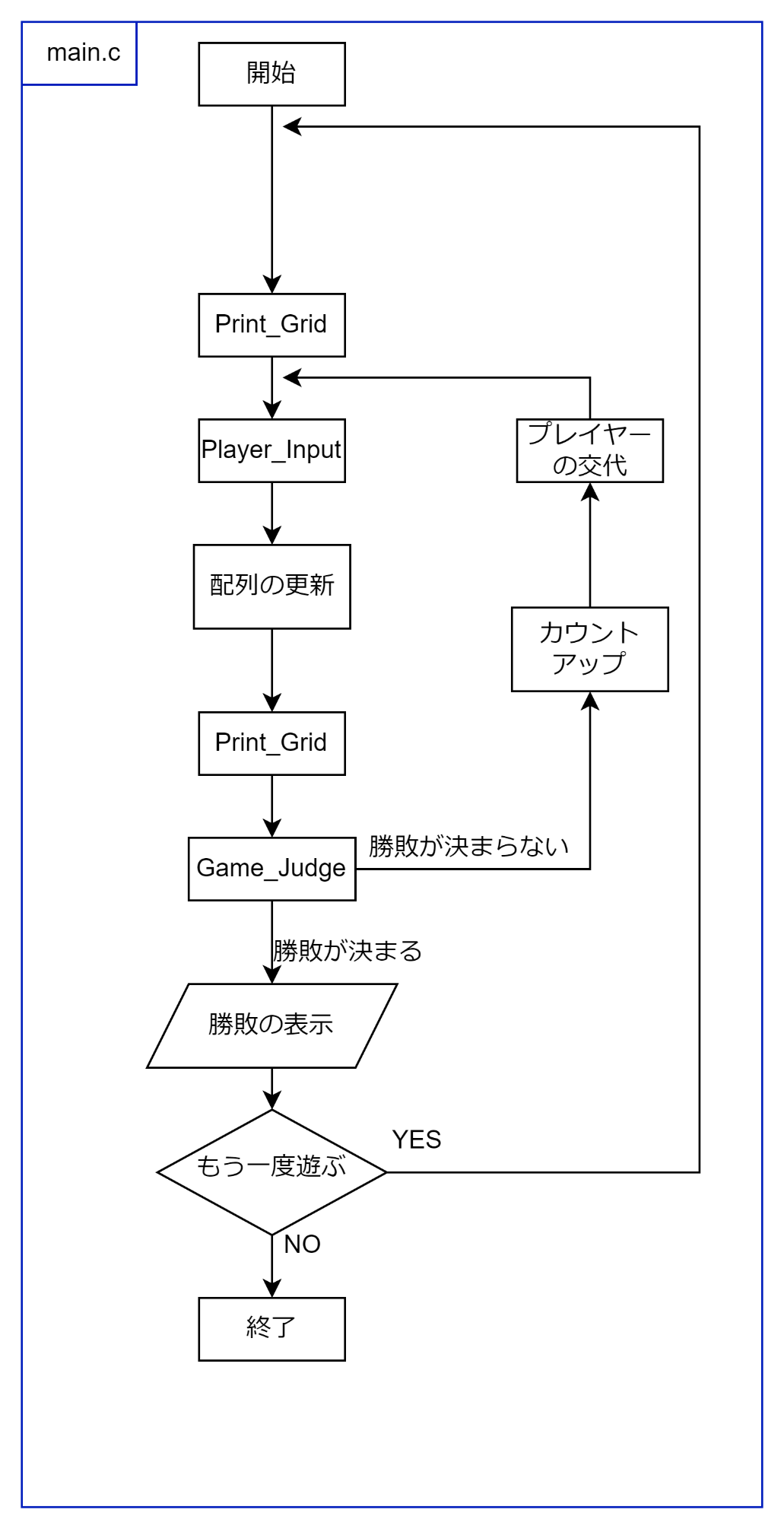
勝敗表示後、リトライ画面を表示する。

入力が‘Y‘の場合はゲームのリトライ、’N‘の場合はゲームを終了する。

ゲームのリトライ時、配列およびカウントはでゲーム開始時と同様に初期化する。

入力が’Y’または’N’以外の場合は、エラーメッセージを表示する。

入力エラー後は、もう一度リトライ画面を表示する。その後、もう一度、リトライできるようにする。



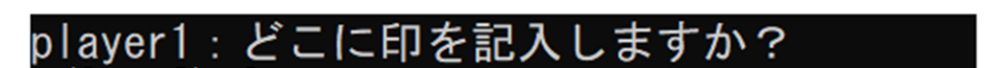
* 1. print\_grid.cファイルの機能

グラフ が含まれている画像

自動的に生成された説明char型3行3列の配列boardを受け取って下図のように[0,0]を原点とするガイド付きの3×3のマスとして表示する。ガイド左上”y.x”は、半角ワイ’y’、半角ドット’.’、半角エックス’x’である。また、記号を表示する際には各マスの中央の半角スペース（左から２番目）をplayer1なら半角オー’o’で、player2なら半角エックス’x’で置き換える。

* 1. player\_input.cファイルの機能
     1. 入力受け取り機能

プレイヤーが指定したマスの座標の入力を、横をｘ、縦をｙで指定し、２回に分けて受け取る。この時、配列では[y][x]となることに留意。また、x＝[0,1,2]、y=[0,1,2]とする。

プレイヤーに入力を促す。

時計 が含まれている画像

自動的に生成された説明ｘの入力を

時計 が含まれている画像

自動的に生成された説明ｙの入力を

と表示して受け取り、構造体BOARD\_XYに格納して返す。

* + 1. 入力受け取りエラー機能

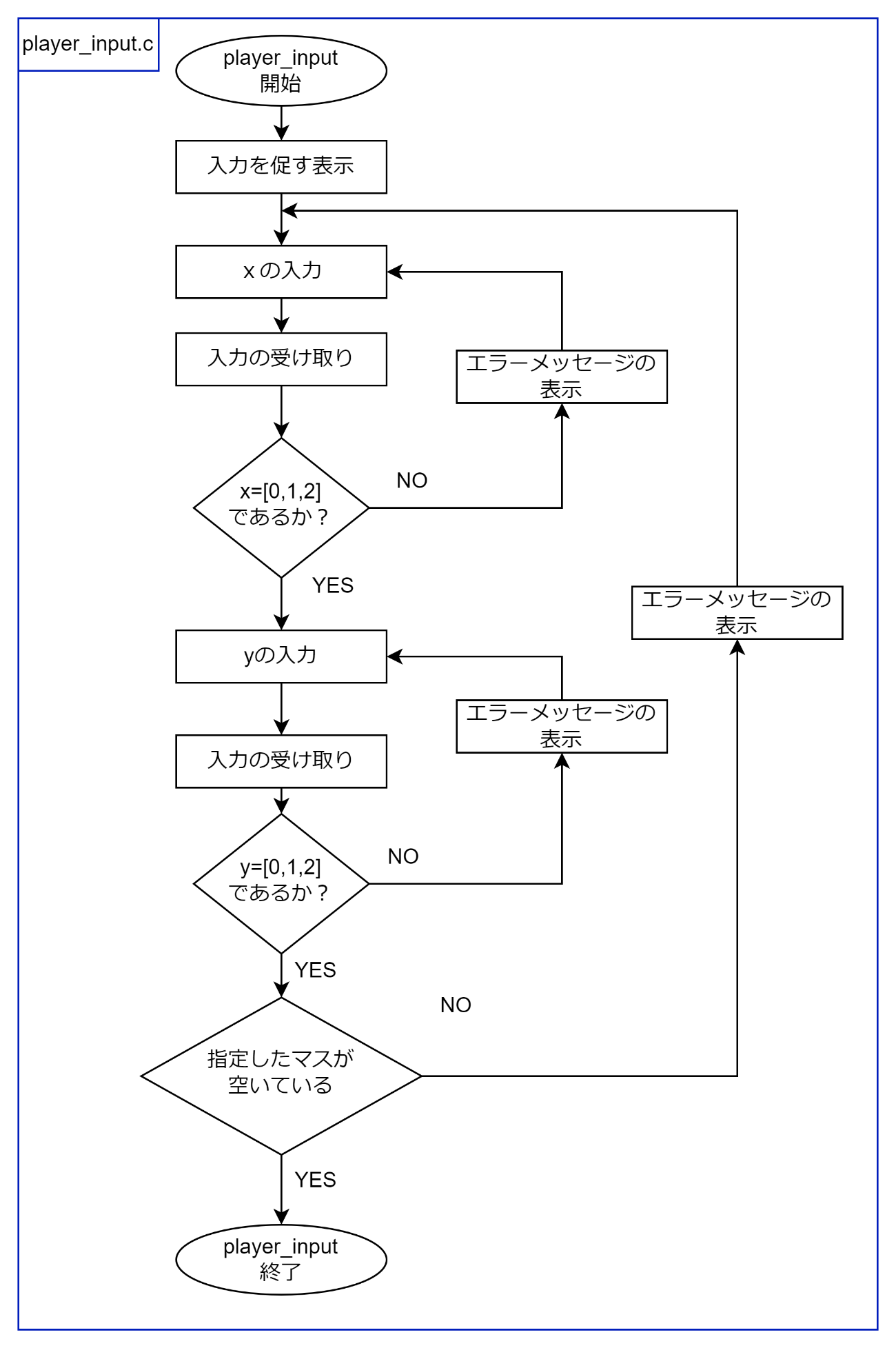
盤面の情報を格納する配列boardの先頭ポインタを参照し、入力は半角数字でのみ受け取るため、範囲外の入力や文字については

とエラーメッセージを表示する。

表示後、xの入力でエラーの際にはｘの入力に、ｙの入力でエラーの際にはｙの入力に戻る。

****半角スペース以外が入力されているマスに入力指示を受けた場合は

とエラーメッセージを表示し、xの入力からやり直す。



* 1. game\_judge.cファイルの機能

ターンのカウントを表す変数とプレイヤーが指定したマスの座標(x,y)を格納する構造体BOARD\_XY、マスの状態を記憶している配列boardを受け取る。まず、4回目のカウントまで揃うことはないので、5回目のカウントから判定を行う。

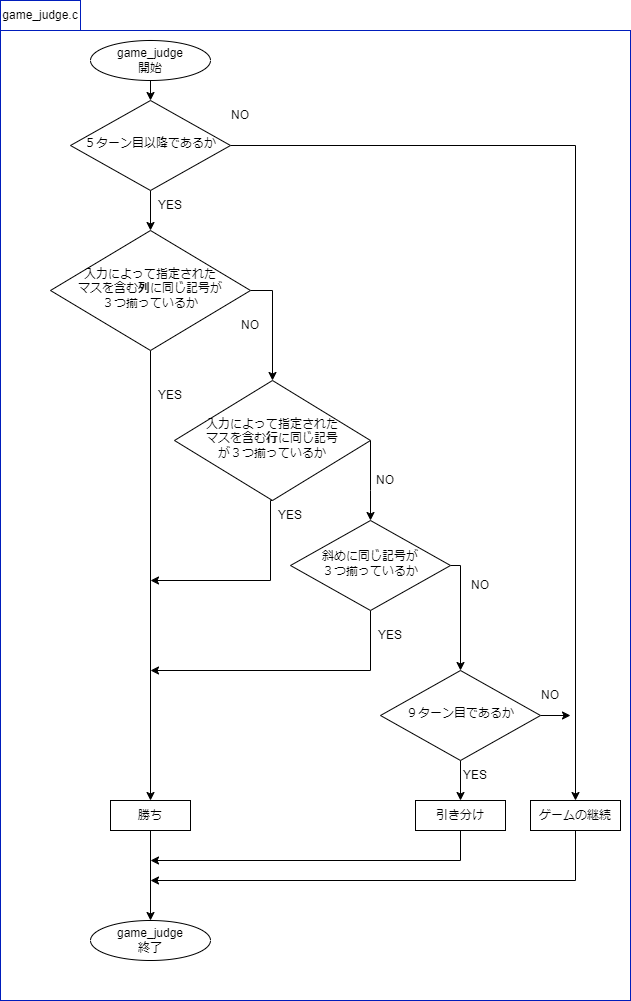
1. 入力によって指定されたマスを含む列のマスを参照する。
2. 入力によって指定されたマスを含む行のマスを参照する。
3. 入力によって指定されたマスが、斜めがそろう条件に関係なくとも斜めのマスを参照する。

これら３つのステップで、同じ記号３つが縦、横、または斜めに揃っているか確認し、揃った場合は勝ちと判定する。カウントが9で、勝敗がつかない場合、引き分けと判定する。1から4回目までのカウントと、勝ちでも引き分けでもないとき、ゲーム継続と判定する。これらの判定結果をRESULT型返り値として返す。

ダイアグラム

中程度の精度で自動的に生成された説明ここで、BOARD\_XYのメンバに0未満の値、3以上の値が入っていた場合、exit関数を用いてプログラムを終了する。同様に、カウントを表す変数に0未満、10以上の値が入っていた場合も、exit関数を用いてプログラムを終了する。

図２：勝敗判定の流れ



# テキスト 低い精度で自動的に生成された説明画面構成

テキスト

自動的に生成された説明テキスト

自動的に生成された説明