内部仕様書

1. プログラムの概要

このゲームはプレイヤーから入力されたプレイ人数をもとにパズルゲームを行う．ゲームクリアはなく，プレイヤーが終了を選択するまで続く．また，ゲームオーバーはあるが，プレイヤーの人数選択画面に戻る．

1. モジュール関連図

* メイン

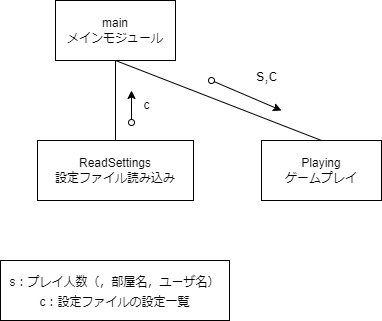


図2.1　メインモジュール関連図

* ゲームプレイ

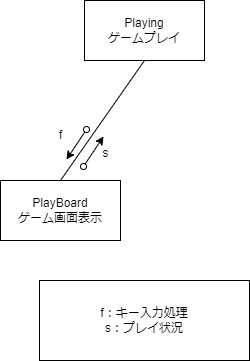


図2.2　ゲームプレイモジュール関連図

* ゲームプレイ
  + ゲーム画面

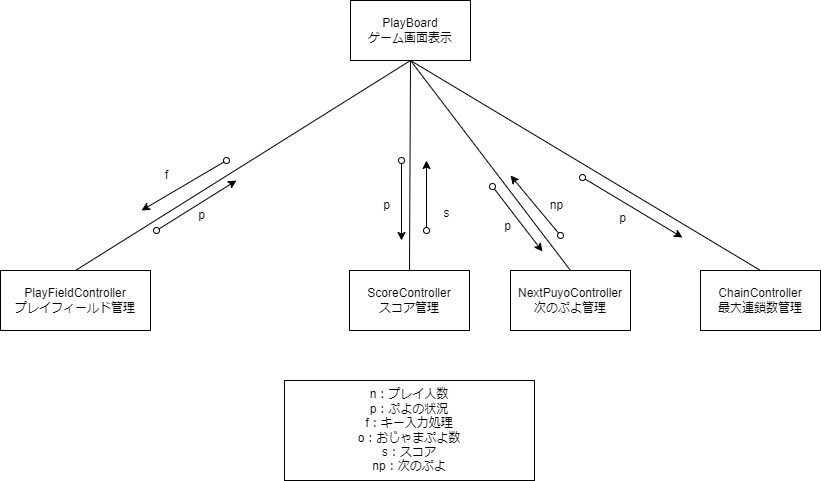


図2.3　ゲーム画面表示・更新モジュール関連図

* ゲームプレイ
  + ゲーム画面
    - プレイフィールド管理

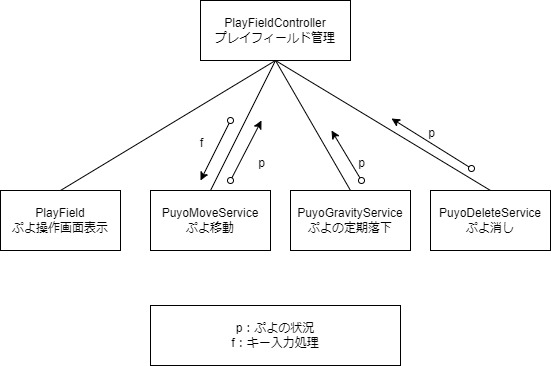


図2.4　プレイフィールド管理モジュール関連図

* ゲームプレイ
  + ゲーム画面
    - スコア管理

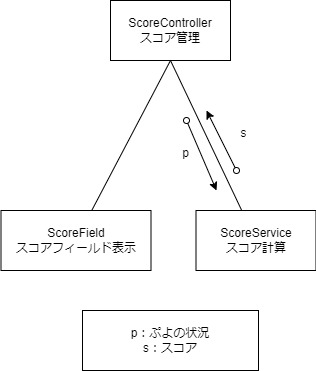


図2.5　スコア管理モジュール関連図

* ゲームプレイ
  + ゲーム画面
    - 次のぷよ管理

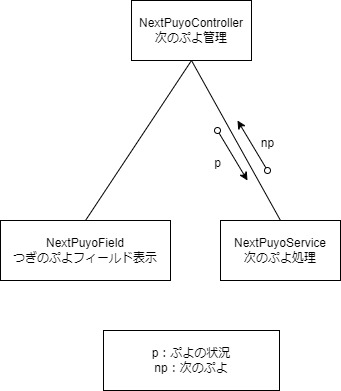


図2.6　次のぷよ管理モジュール関連図

* ゲームプレイ
  + ゲーム画面
    - 最大連鎖数管理

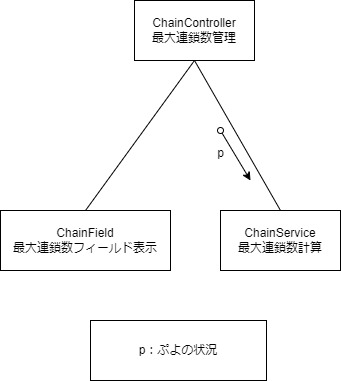


図2.7　最大連鎖数管理モジュール関連図

1. モジュール流れ図

* メイン

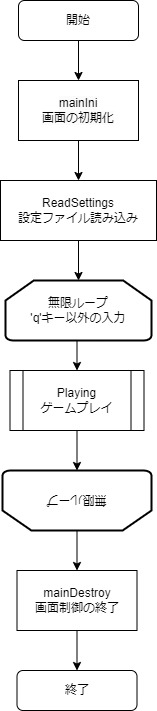


図3.1　メインモジュール流れ図

* ゲームプレイ

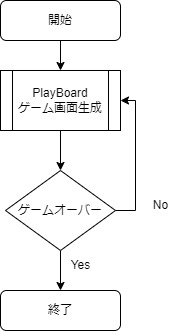


図3.2　ゲームプレイモジュール流れ図

* ゲームプレイ
  + ゲーム画面

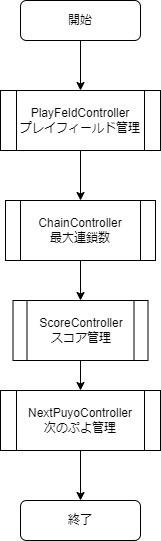


図3.3　ゲーム画面表示・更新モジュール流れ図

* ゲームプレイ
  + ゲーム画面
    - プレイフィールド管理

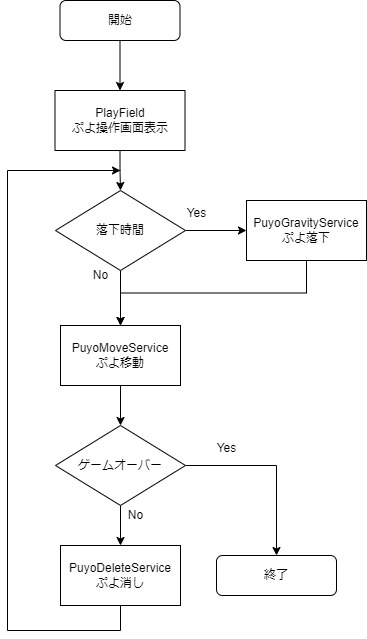


図3.4　プレイフィールド管理モジュール流れ図

* ゲームプレイ
  + ゲーム画面
    - スコア管理

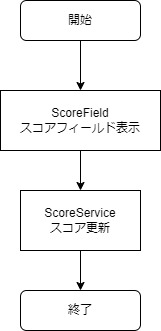


図3.5　スコア管理モジュール流れ図

* ゲームプレイ
  + ゲーム画面
    - 次のぷよ管理

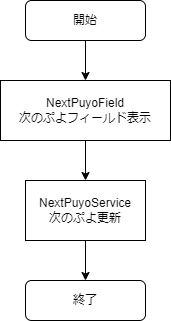


図3.6　次のぷよ管理モジュール流れ図

* ゲームプレイ
  + ゲーム画面
    - 最大連鎖数管理

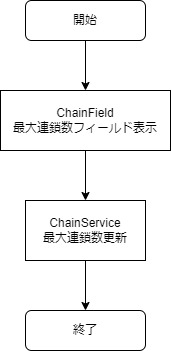


図3.7　最大連鎖数管理モジュール流れ図

1. 関数仕様一覧

* メイン





* 設定ファイル読み込み







* ゲーム開始設定クラス



|  |  |  |  |
| --- | --- | --- | --- |
| 関数名 | PlaySettings::settings | | |
| 引数 | なし | 戻り値 | 各種設定 |
| 概要 | 各種設定を入力してもらう | | |
| 処理内容 | プレイ人数の選択からプレイ開始までに必要な設定を行う 複数人プレイの場合はプレイ人数を満たすまで待機する | | |

* プレイ人数設定





* 画面サイズ検証





* 画面サイズ例外



* ゲームプレイ





* ゲーム画面管理

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 関数名 | PlayBoard::PlayBoard | | | | |
| 引数 | 型 | 変数名 | 概要 | 戻り値 | なし |
| const int | y | フィールドの y 座標 |
| const int | x | フィールドの x 座標 |
| struct settings | settings | 各種設定 |
| 概要 | PlayBoardクラスのコンストラクタ | | | | |
| 処理内容 | PlayFieldControllerクラス ChainControllerクラス ScoreControllerクラス NextPuyoControllerクラス のインスタンスをメンバ変数に格納する | | | | |



* 操作画面管理

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 関数名 | PlayFieldController::PlayFieldController | | | | |
| 引数 | 型 | 変数名 | 概要 | 戻り値 | なし |
| const int | y | フィールドの y 座標 |
| const int | x | フィールドの x 座標 |
| struct settings | settings | 各種設定 |
| 概要 | PlayFieldControllerクラスのコンストラクタ | | | | |
| 処理内容 | 親クラスのDynamicFieldのコンストラクタを呼び出してメンバ変数を格納する ぷよを操作するフィールドを表示する． PuyoMoveServiceクラスとPuyoGravityServiceクラスのコンストラクタを メンバ変数に代入する プレイヤーが操作するぷよ2つの初期座標をメンバ変数に代入する | | | | |





* ぷよ操作機能









* ぷよ落下機能

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 関数名 | PuyoGravityService::gravity | | | | |
| 引数 | 型 | 変数名 | 概要 | 戻り値 | なし |
| std::array<std::array<Puyo, 10>, 6>& | puyos | フィールド上のぷよ |
| 概要 | 一定時間ごとにぷよを1マス落下させる | | | | |
| 処理内容 | 一定時間ごとにぷよを落下させる | | | | |

* ぷよ削除機能



* プレイ画面













* 連鎖画面管理











* 連鎖数機能









* 最大連鎖数画面















* スコア管理











* スコア機能







* スコア画面



















* 次のぷよ管理











* 次のぷよ機能





* 次のぷよ画面











* ぷよ















1. 画面設計

実際のゲーム画面のイメージを以下に示した．また，このゲームでは画面の大きさを20×50以上とする．

座る, 画面, コンピュータ, 操作 が含まれている画像

自動的に生成された説明

図5.1　ゲーム画面

1. 変更の履歴

7/20 ・ 複数人数での対戦関連のモジュール関連図を削除

・ 複数人数での対戦関連の流れ図を削除

・ 複数人数での対戦関連の関数仕様を削除

・ 複数人数での対戦関連の設定画面を削除

・ プログラム作成に伴い実装不可能な関数部分を変更