

# Camp Game (陣取りゲーム)

## 標準仕様

- プレイヤー(入力) 対 プレイヤー(入力)
- マスは3x3以上
- 先攻と後攻をランダムで決定
- プレイヤーは座標入力で石を置く
- すでに石が置かれている座標に石を置くことはできない
- 置くことができない座標が指定された場合は再度入力を求める
- 縦・横・斜めのいずれかが一直線になったら勝利とする(勝利した側の名前も出力)
- マスが全て埋まってしまったら引き分けとする
- ターミナルにマス目の出力
- 自然な座標表記(0,1,2 -> 1,2,3)

## テスト項目

- 1ターンおきにプレイヤーが正常に入れ替わるか
- 同じ座標に石を置いてエラー処理が行われるか
- マイナスの座標等を入力してエラー処理が行われるか
- 縦・横・斜めのいずれかが揃った際に正常にゲームが終了するか
- 勝敗が決することなくマスが全て埋まった場合、引き分けとして正常にゲームが終了するか

## 発展

- プレイヤー(入力) 対 プレイヤー(CPU)
- CPUは任意のアルゴリズムに従って石を置く(ランダムを避けるのが望ましい)
- 絶対に負けないアルゴリズムを搭載したCPUの作成
- スタート時に対人モードと対CPUモードを選択可とする

## 構成

CampGame -> Func -> Scene -> Player -> Move -> Check

### CampGame

- main()
  - area ... 現在のエリア状況
  - playerNum ... プレイヤーの識別番号
  - turn ... ターン数
  - mode ... 選択されたモード

### Func

- getRandPlayerNum() ... 先攻を決める乱数を取得
- getMode() ... モードを取得
- changeTurn() ... ターンを更新し出力
- changePlayerNum() ... ターンの更新に合わせてプレイヤーを更新

### Scene

- clearScene() ... 画面のリフレッシュ
- startScene() ... スタート画面
- gameScene() ... ゲーム画面
- resultScene() ... リザルト画面

### Player

- updateArea() ... エリアの更新
  - human() ... プレイヤー(入力)の場合の動作
  - cpu() ... プレイヤー(CPU)の場合の動作

### Move

- putBlankPoint() ... ブランクに置く
- putDirected() ... 入力座標に従って置く
- putCenter() ... 中心に置く
- putCorner() ... 角に置く
- putLinear() ... 角以外に置く
- putReachPoint() ... リーチの座標に置く
- putWinningPoint() ... 勝利が確定する場所に置く

### Check

- center() ... 中心がブランクか
- reach() ... リーチがあるか
- win() ... 勝利条件が満たされているか
- end() ... ブランクがあるか