

第2回 SASプログラミングコンテスト リバーシーチャンピオンシップ



SASユーザー総会 2026



開催概要

- 今年もリバーシーチャンピオンシップを開催します!!
- ルールは昨年と一緒ですが、制限時間を延長しています
- 公開されているプログラムの流用はNGです
機械的および視覚的に類似性を確認します
類似度の高い人には出場を辞退いただきます
- 昨年同様に配布資料を利用してプログラム開発ください
(提供プログラムは使いやすいように各々改変ください)
- 不明点は井上貴博(takahiro.inoue.y1@kyowakirin.com)まで
- 配布資料
 - ▣ プログラム開発用対戦プログラム: platform_v1.txt
 - ▣ 対戦用サンプルプログラム: tester_v1.txt
 - ▣ 自己紹介スライド(テンプレート): Self-intro.pptx

プログラム・参加条件

コンテスト参加・実装ルール

- ・ コンテスト実施時は、会場でご参加ください。
- ・ プログラムは マクロで作成します。マクロ名は世話人より指定します。例: INOUEA1(姓5文字 + 名2文字 + 連番)
- ・ 出場者のマクロを SASユーザー総会 公式Webサイト に公開することに同意してください。
- ・ SAS学習者の参考となるように、コメントを適宜追記し、分かりやすい構文を心掛けてください。

盤面データ・入出力仕様

- ・ 盤面データとして、SAS データセット[_info]を読み込み、選択したセルをグローバルマクロ変数[_xy]に x,y の形式で格納してください。例: [X=7, Y=3]: 7,3
- ・ 対局開始時に黒または白の担当が決定されます。どちらの担当でも動作するプログラムを作成してください。
- ・ 右図の場合、17手目 (No=17) で Turn = -1 (黒)の指し手となります。変数 Y, X1-X8 には、16手目までの盤面情報が格納されており、黒は -1、白は 1 で表現されます。17手目で指せるセルには 99 が入力されています。選択可能セル (99) から1つを選択し、_xy に設定してください。

命名・使用制限

- ・ マクロ内で、アンダースコア (_) から始まるデータセット名・マクロ変数名は作成しないでください。
- ・ 文字変数の入力値はASCII文字のみを使用してください。コメントには日本語等の使用は可能です。
- ・ 対戦プログラムでは[_info]以外にもデータセットが作成されますが、使用できません。コンテスト当日は、ライブ対戦用に別プログラムを使用するため、[_info]を除いたデータは名称・構造ともに異なります。

対局・実行条件

- ・ 動作不良は負けとなります。対局は、対戦者同士のマクロを 交互に実行します。細かな命名規則は定めませんので、データセット名の重複に注意してください。
- ・ SAS OnDemand 実行時に 1手6秒以内 (7秒未満) を目安としてください (合計時間ではありません) 。

Length No Y Turn X1 - X8 8. ;
No: ○手目
Turn: 指し手の色
(黒 -1、白 1)
Y: 上から○マス(行)目
(Y=1は1行目、Y=2は2行目...)
X1: 左から1マス(列)目
(黒 -1、白 1、選択可能セル 99)
X2~X7: 左から2~7マス(列)目
X8: 左から8マス(列)目

提供データ(_info.sas7bdat)

No	Turn	Y	X1	X2	X3	X4	X5	X6	X7	X8
17	-1	1	0	0	0	0	0	0	0	0
17	-1	2	0	0	0	0	99	99	0	0
17	-1	3	0	0	-1	-1	1	1	99	0
17	-1	4	0	0	-1	-1	1	99	99	99
17	-1	5	0	-1	-1	1	1	1	1	1
17	-1	6	0	99	1	-1	-1	-1	99	0
17	-1	7	0	0	99	99	-1	1	99	0
17	-1	8	0	0	0	0	0	99	99	0

<イメージ>

Turn	Y	X1	X2	X3	X4	X5	X6	X7	X8
●	1								
●	2					99	99		
●	3			●	●	○	○	99	
●	4			●	●	○	99	99	99
●	5		●	●	○	○	○	○	
●	6		99	○	●	●	●	99	
●	7			99	99	●	○	99	
●	8						99	99	

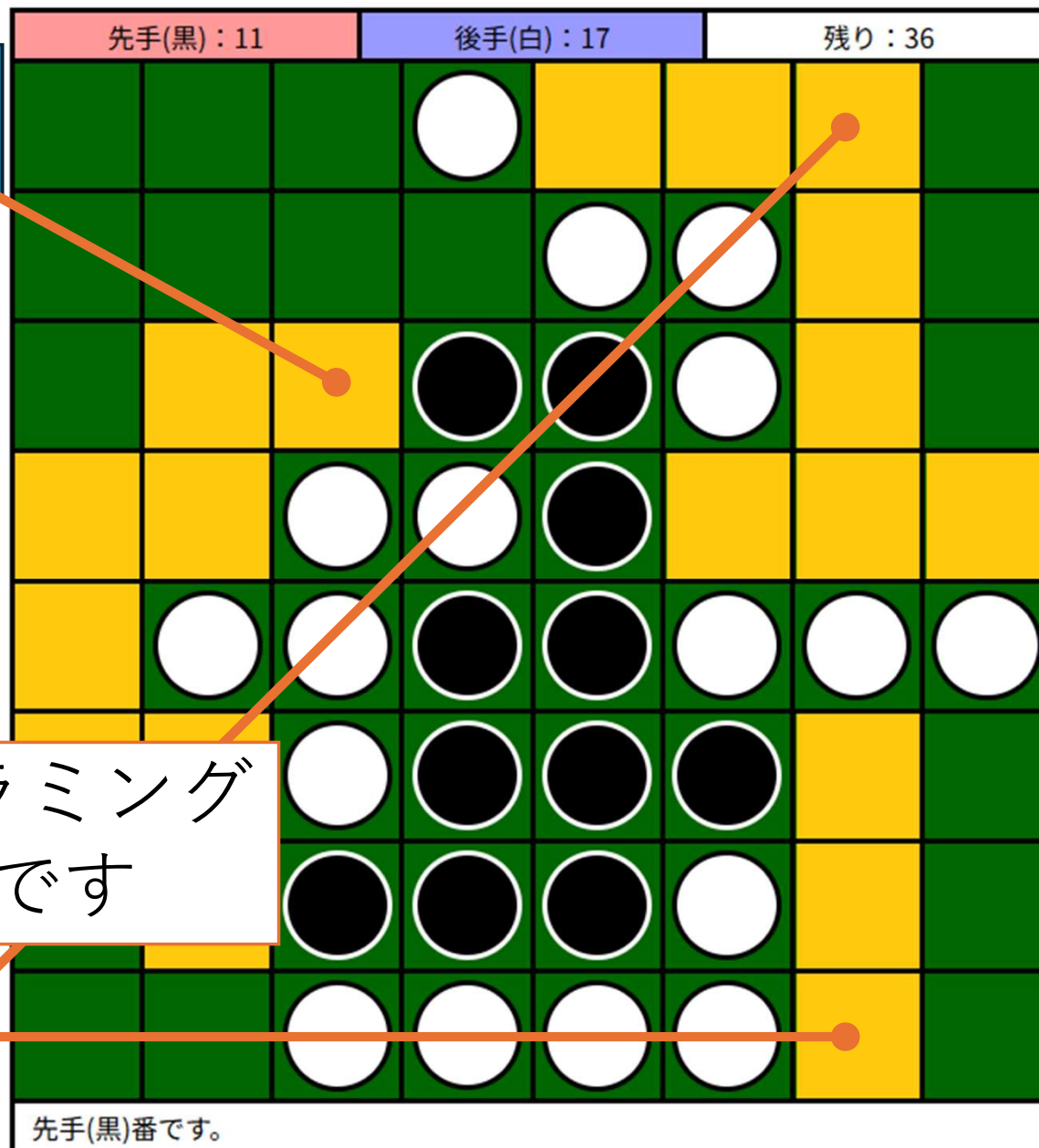
一番多く反転するけど
次で戻される

先手(黒)です
どこに置く？

黄色が置ける
ところ

こういった思考をプログラミング
強い人がチャンピオンです

ここに置くと角が
取られやすい？



公開プログラムとの類似性

	IIJIMTA2	SEKINSH2	IZUMITO2	HIRAIKE1	IIJIMTA1	IIJIMYA1	IZUMITO1	KAWAKTA1	OTAYU1	SEKINSA1	SEKINSH1
IIJIMTA2	100.0%	40.9%	39.0%	55.4%	62.7%	19.6%	27.8%	53.0%	27.7%	21.5%	54.6%
SEKINSH2	40.9%	100.0%	49.7%	41.4%	41.5%	22.9%	36.4%	39.2%	22.1%	23.6%	42.6%
IZUMITO2	39.0%	49.7%	100.0%	42.0%	41.8%	23.4%	37.0%	42.0%	22.6%	21.0%	48.2%
HIRAIKE1	55.4%	41.4%	42.0%	100.0%	57.2%	22.7%	30.0%	54.1%	26.3%	25.0%	54.2%
IIJIMTA1	62.7%	41.5%	41.8%	57.2%	100.0%	20.9%	30.0%	51.5%	28.5%	22.6%	57.3%
IIJIMYA1	19.6%	22.9%	23.4%	22.7%	20.9%	100.0%	21.5%	22.6%	21.2%	37.4%	22.1%
IZUMITO1	27.8%	36.4%	37.0%	30.0%	30.0%	21.5%	100.0%	29.7%	22.6%	22.8%	31.8%
KAWAKTA1	53.0%	39.2%	42.0%	54.1%	51.5%	22.6%	29.7%	100.0%	29.1%	23.6%	53.5%
OTAYU1	27.7%	22.1%	22.6%	26.3%	28.5%	21.2%	22.6%	29.1%	100.0%	24.3%	28.5%
SEKINSA1	21.5%	23.6%	21.0%	25.0%	22.6%	37.4%	22.8%	23.6%	24.3%	100.0%	23.8%
SEKINSH1	54.6%	42.6%	48.2%	54.2%	57.3%	22.1%	31.8%	53.5%	28.5%	23.8%	100.0%

流用が疑われる出場者はいませんでした。

類似度を定量化した手法は、以下の発表を参照

SASプログラムの類似度評価 ～Token化と5-gramによる構造比較～

対戦形式

- 3名による総当たり戦
- 各組合せで先後を入れ替えて2戦
- 先手：黒 ／ 後手：白
- 全3カード、計6試合
- 制限時間内に応答がない場合は負け
- 勝ち：3点 ／ 引き分け：1点 ／ 負け：0点
- 合計ポイントで順位を決定

SASユーザー総会 2026
SASプログラミングコンテスト

リバーシーチャンピオンシップ

IZUMITO2

お名前（ご所属）	伊住 寿雄（ソーバル株式会社）
プログラムの特徴	• $\alpha\beta$ 探索法を用いて最大15手までの先読みを行う。（ただし作成環境では8～10手が限界） • 制限時間を超えそうな場合、探索を途中で打ち切り、打ち切り時点での最善手を選択する。 • 探索中は、「石を置く位置」、「角からの確定石（今後反転しない石）の数」、「石の差」、「すでに置いた石の価値」、「石を置ける場所の数」の観点で探索局面の評価値を算出し、残りマス数に応じて各評価の重みを変えるようにしている。 • サブルーチンおよびarray_temporary_を用いて、ほとんどの計算をメモリ上で行うようにした。そのため、1データステップで実行できるようになった。（データステップの読み書きを削減） • 探索中の盤面を1手前に戻す作業は、盤面全体をコピーするのではなく、変更した箇所のみを行うことで処理手数を抑え、探索を高速化した。（Undo方式） • 探索時は1階層手前（先読み時のn-1手）での最善手から探索を行うようにして、無駄な探索を減らすようにした。
AI利用について	AI利用については、 $\alpha\beta$ 探索法のアルゴリズムやプログラム構成を提案、作成したコードを読み込ませてのデバッグ、残りマス数に応じた評価関数の重みづけのおすすめの割合の算出等について利用した。
コメント	探索深度がPCの処理速度に依存しているため、強さは実行環境次第。とにかく処理速度と探索深度重視です。昨年よりは強いものができたと思っています。目指せ優勝！

SASユーザー総会 2026
SASプログラミングコンテスト

リバーシーチャンピオンシップ

IIJIMTA2

お名前（ご所属）	飯島 泰平(ネイチャーインサイト株式会社)
プログラムの特徴	5手先の多くの盤面を探索し、それぞれの盤面に点数付けし、Minimax法で双方最善を想定した下での現在の盤面の着手を決める 点数付けの観点：相手の選択肢の少なさ、中央付近の自分の石の多さ、確定石(終局まで返されない石)の多さ、X打ち(隅の斜め内側)の自分の石の少なさ、現在の盤面から相手の選択肢を増やさない手を優先、など 最終盤の残り8手は終局盤面まで探索して、点数付けを自分の石数にすることにより、(パスを含む手順を除いて)最善手を打てる
コメント	今年も開催いただきありがとうございます。 昨年より制限時間が伸びたことにより、より深く読めるようになったため、より強いプログラムを作成できたと思います。私自身がこのプログラムと対局して負けることも多かったです(笑) ただし、昨年のプログラムに勝てたものの、大差ではなかったため、本番では厳しい内容になる気がしています。 対局よろしくお願いします。

SASユーザー総会 2026
SASプログラミングコンテスト

リバーシーチャンピオンシップ

SEKINSH2

お名前（ご所属）	関根伸吾（ネイチャーインサイト株式会社）
プログラムの特徴	前回のプログラムは、盤面のコマ数や着手可能数などの特徴量をもとに局面を評価し、ミニマックス法で約4～6手先まで読むロジックでした。 今回は評価指標をさらに拡充し、確定石や危険領域の評価など合計20種類の特徴量を導入しました。さらに、ゲーム進行（序盤・中盤・終盤）に応じた3つの基準点を設定し、その間を区分線形補間する仕組みを構築しました。これにより、局面の推移に合わせて評価の重みを滑らかに変化させ、状況に応じた最適な判断を可能にしています。 このプログラムは、Claude Codeを用いてベースとなるSASプログラムを生成し、そこから修正を加えて完成させております。
コメント	前回より強くなったと思いますので、ぜひ勝ちたいです！

チャンピオン: 関根伸吾 (ネイチャーインサイト株式会社)

		伊住寿雄 IZUMITO2	飯島泰平 IIJIMTA2	関根伸吾 SEKINSH2	合計 ポイント
		後手: 白 先手: 黒	後手: 白 先手: 黒	後手: 白 先手: 黒	
伊住寿雄 IZUMITO2	先手: 黒		3	0	6
	後手: 白		3	0	
飯島泰平 IIJIMTA2	先手: 黒	0		0	0
	後手: 白	0		0	
関根伸吾 SEKINSH2	先手: 黒	3	3		12
	後手: 白	3	3		

勝ち：3点 / 引き分け：1点 / 負け：0点

Backup

エキシビションマッチ

開発主体：AI (Microsoft Copilot)

開発時間：約1.5時間(90分)-1): COPILOT / 白(1): IJIMTA1

人間によるコーディング：0行

人間によるデバッグ：0行

人間による設計判断のみ

	X1	X2	X3	X4	X5	X6	X7	X8
	●	●	●	●	●	●	●	○
2	●	●	○	○	●	●	●	○
3	●	○	●	○	○	●	●	○
4	●	○	●	●	●	●	○	○
5	●	○	●	●	●	○	○	○
6	●	●	○	●	○	●	○	○
7	●	●	●	○	○	○	○	○
8	●	●	●	●	●	●	○	●

黒(-1): IJIMTA1 / 白(1): COPILOT

y	X1	X2	X3	X4	X5	X6	X7	X8
1	○	○	○	○	○	○	○	○
2	○	○	●	●	●	●	●	○
3	○	○	○	○	○	○	●	○
4	○	○	●	○	○	●	●	○
5	○	○	○	○	○	●	○	○
6	○	○	○	○	○	○	●	○
7	○	○	○	●	●	●	○	○
8	○	○	○	○	○	○	○	○

黒(-1): COPILOT / 白(1): IJIMTA1

FREQ プロシジャ

COL1	度数	パーセント
●	39	60.94
○	25	39.06

黒(-1): IJIMTA1 / 白(1): COPILOT

FREQ プロシジャ

COL1	度数	パーセント
●	14	21.88
○	50	78.13

SASユーザー総会 2026
SASプログラミングコンテスト
リバーシーチャンピオンシップ

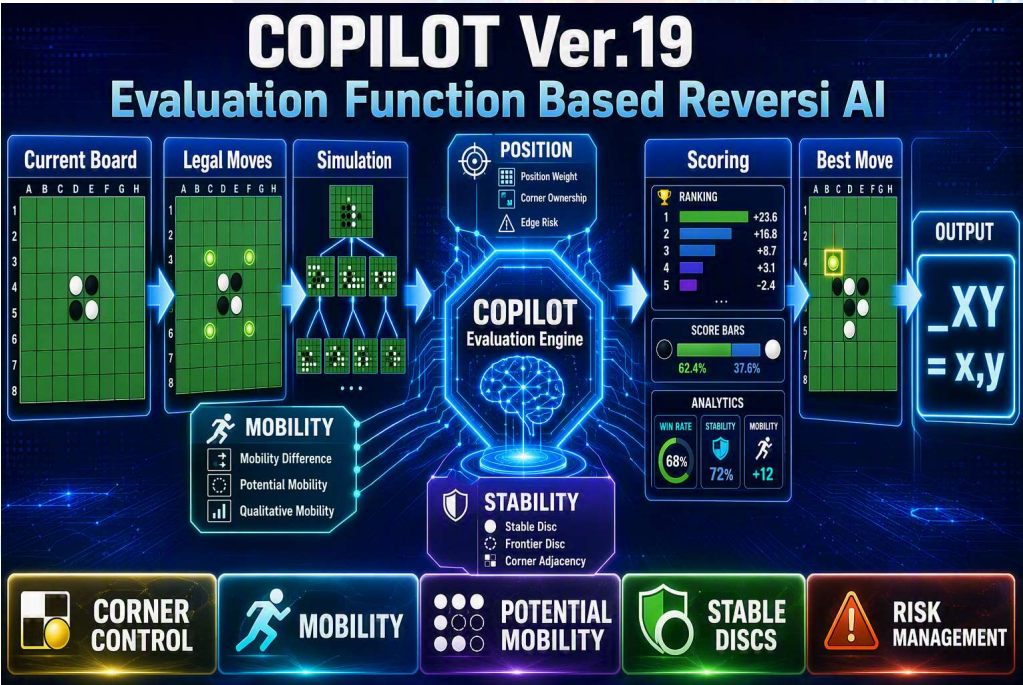
COPILLOT

プログラムの
特徴

各合法手について着手後の盤面をシミュレーション
着手後局面を以下の観点で評価

- Position Weight(位置評価)
- Mobility Difference(合法手差)
- Potential Mobility(将来の合法手候補)
- Qualitative Mobility
(角・辺・内部の合法手分類)
- Stable Disc(安定石)
- Corner Ownership(角保有数)
- Dynamic Corner Adjacency
(角取得後の隣接マス評価)
- X-square / C-square Risk
- Frontier Disc
- Edge Risk

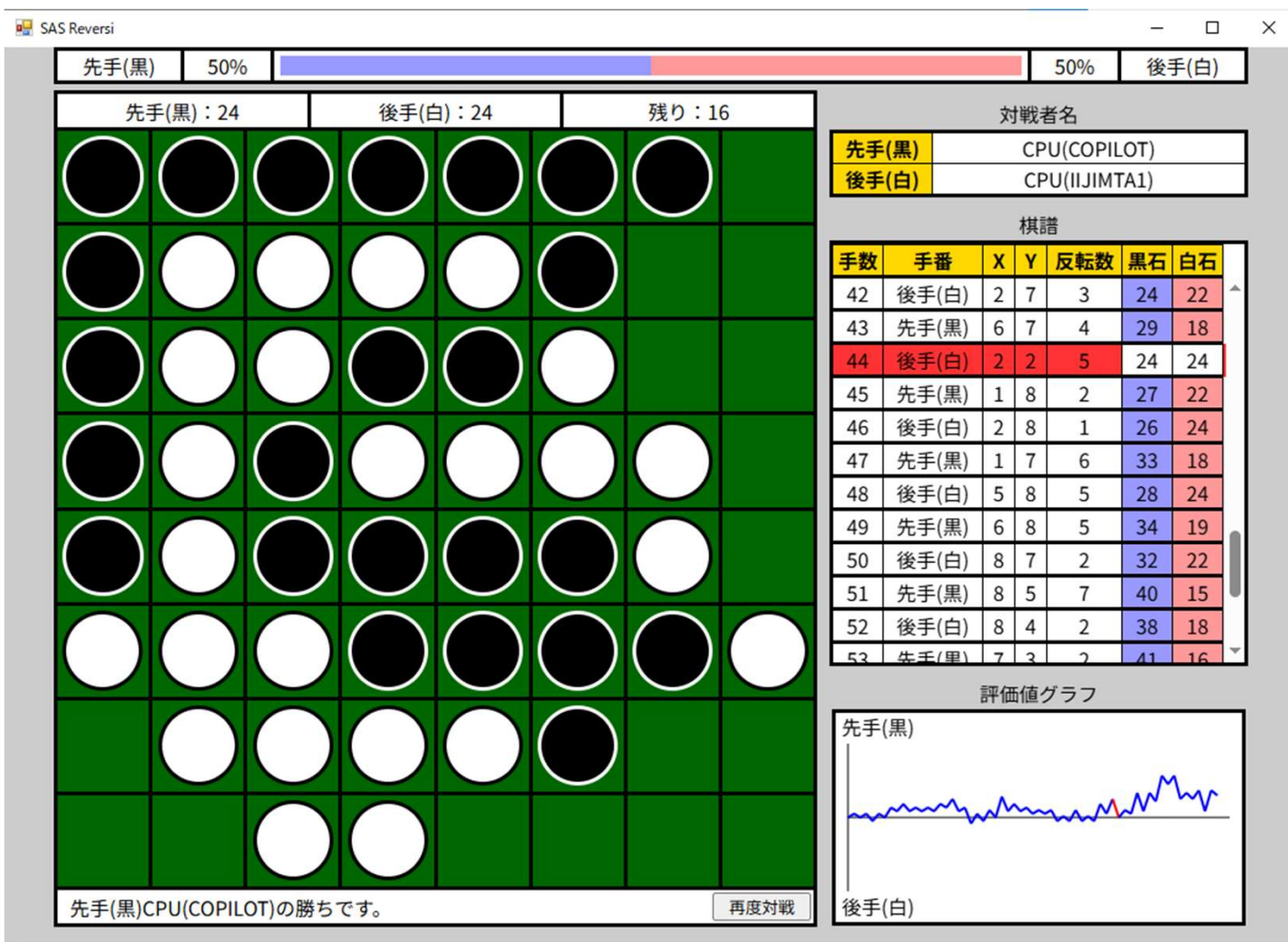
序盤・中盤・終盤で評価係数を切り替え
最も評価値の高い手を選択



コメント

全てCopilotにプログラミングさせました。
作者は指示のみで、まったくプログラミングしていません。

対戦プラットフォーム



2024年 SASクリエイティブ賞
伊住寿雄氏（ソーバル株式会社）
より提供いただきました。

詳細は2024年の資料をご参照ください

無料配布してくださるとのこと
ご興味がある方は
toshio_izumi@sobal.co.jp
までご連絡ください。

SAS Programming Contest Reversi Championship

This is an algorithmic programming competition focused on Reversi.

Participants develop decision-making logic using SAS programming. Each program determines where to place a coin based on the current board state, and competes automatically against other players' algorithms.

This competition focuses on strategy and algorithmic thinking. No user interface or graphics implementation is required.

Participants from all countries are welcome.



If you wish to participate, please contact Takahiro Inoue (takahiro.inoue.y1@kyowakirin.com) by 20Aug2026. For participants joining from outside Japan, on-site participation may be difficult; therefore, the match results and videos will be shared with participants at a later date. Please note that reuse of publicly available programs is not permitted. Program development must be based on the distributed materials listed below. If you have any questions, please contact Takahiro Inoue.

Distributed materials

- Match program template for development: platform_v1.txt
- Sample program for matches: tester_v1.txt
- Self-introduction slide (template): Self-intro.pptx

Reversi – Basic Rules

- Reversi is played on an 8×8 board by two players: Black and White
- The game starts with four coins in the center
- Black moves first
- A move is valid if it sandwiches one or more opponent coins in a straight line (horizontal, vertical, or diagonal)
- All sandwiched coins are flipped to the current player's color
- If a player has no valid moves, they must pass
- The game ends when neither player can move
- The player with more coins on the board wins

Programming Requirement

General Requirements

- The program must be implemented as a SAS macro.
- The macro name will be specified by the organizers.
- By participating, you agree that your macro code may be published on the official SAS User Group website.
- Please write clear and well-commented code so that it is helpful for SAS learners.
Use ASCII characters only in code and input values (comments may include other languages)

Input / Output Specification

- The current board state is provided in a SAS dataset named `_info`.
- Your program must select one valid cell and output the move by setting the global macro variable `_xy` in the format: `x,y` [e.g., (X=7, Y=3): 7,3]

Player Color

- At the start of a match, your program is assigned Black or White.
- Your macro must work correctly for both colors. The assigned color does not change during a match.

Board Representation

- Variables `Y`, `X1`–`X8` represent the board state up to the previous move. Cell values: [-1 : Black] [1 : White] [99 : Valid move positions]
- Your program must choose one cell with value 99 as the next move.

Turn Information

- The variable `No` indicates the move number. The variable `Turn` indicates the current player [e.g., Black = -1, White = 1].

Naming Restrictions

- Do not create any dataset names or macro variables that start with an underscore (`_`).

Data Usage Restrictions

- Other datasets may exist during match execution, but only `_info` may be used.
- During live matches, datasets other than `_info` may differ in name and structure.

Execution and Stability

- If your macro fails to execute correctly, the match is considered a loss.
- Macros from both players are executed alternately during a match.
- Dataset name conflicts may cause unexpected behavior.

Time Limits

- As a guideline, execution time should be under 6 seconds per move when run on SAS OnDemand.
- The limit applies per move, not to the total match duration.

Input: `_info.sas7bdat`

No	Turn	Y	X1	X2	X3	X4	X5	X6	X7	X8
17	-1	1	0	0	0	0	0	0	0	0
17	-1	2	0	0	0	0	99	99	0	0
17	-1	3	0	0	-1	-1	1	1	99	0
17	-1	4	0	0	-1	-1	1	99	99	99
17	-1	5	0	-1	-1	1	1	1	1	1
17	-1	6	0	99	1	-1	-1	-1	99	0
17	-1	7	0	0	99	99	-1	1	99	0
17	-1	8	0	0	0	0	0	99	99	0