部　　門	競　技　部　門	No.1 登録番号	30001

No.2

1) 予定開発期間：７ヶ月
2) 予定開発人数：３人

	４月			５月			６月			７月			８月			９月			１０月		
問題分析	←														→						
設計	←																→				
実装				←															→		
試用・トレーニング							←												→		

No.3

実現方法

1) フィールド上のペアを多く作るアルゴリズム

本競技では、最終的にフィールドを 4×4 の領域にまで問題を縮小できれば、全探索によって解を導くことが可能である。全探索により、すべてのペアを揃えることが理論上可能であるため、まずは大きなフィールドを段階的に縮小し、最終的に 4×4 に帰着させることを目指す。

考察したアルゴリズムを以下に示す。

1. 図１のように左上の角から横方向に、同じ数字がペアになっているエンティティ２つで構成される $2 \times m$ の部分(以下「ブロック」と呼ぶ)を幅優先探索を用いて揃える。図では $m = 2$ の場合について示す。
2. 1.で揃えたブロックの右側に、同様の手順でブロックを $N - m$ 列目まで揃えていく(図 2)。
3. 上２行を完全に揃えるために、右上のブロックを整える際は、図３に示すように、端から m マスを空けて下側で揃える(オレンジ色の部分)。この操作によって、図４のように $(m + 2) \times (m + 2)$ の園を導くことで、上２行全体が完全に揃う状態になる。また、この操作により図３のオレンジ色の部分の上にある数字のペアが崩れないよう、事前にペアの向きを調節して揃えておく必要がある。
4. 続いて、右２列についても、1.~3. の方法で順に整えていく(図５、図６)。このときは、縦と横を交換した操作を適用することになる。
5. 最終的に図６のように上２行と右２列が揃い、外枠が完成する。

この時点で、まだ整列されていない領域に注目すると、元の $N \times N$ のフィールドは、$(N - 2) \times (N - 2)$ のサイズへと縮小されていることがわかる。したがって、この縮小された領域にも同じ手法を適用すればよい。

このようにして、フィールドを段階的に縮小しながら、再帰的に整列を進めていくことで、4×4 のフィールドに帰着させることができる。その段階で全探索を行うことで全てのペアを揃えることが可能となる。

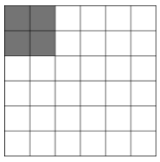
図１

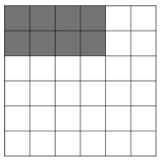
図２

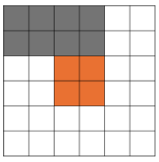
図３

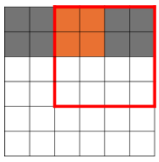
図４

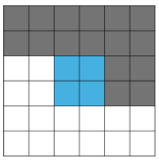
図５

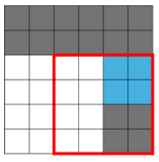
図６

2) 導きの手数を少なくする工夫

前述の解法においては、外枠を揃える際に、各操作でより多くのペアを同時に形成する手法を取ることで、全体の手数を削減することが可能となる。特に有効な手法の一つとして、ビームサーチを用いる方法が挙げられる。これにより、短時間で効率的な操作手順の選択できることが期待できる。

ビームサーチによる効果を最大限に引き出すためには、高速化も重要な要素である。そのため、少ない手数でブロックを揃えることを目指しつつ、探索途中で明らかに揃う見込みのない盤面を枝刈りするなどの工夫が有効である。

これらの工夫を組み合わせることで、限られた時間内に、より少ない手数で全てのペアを揃えることが可能となる。

3) その他（独創的なところ）

実装には OpenSiv3D を用いており、ビジュアルによる動作確認や直感的なインターフェースの構築が容易である。また、処理の高速化を目的として並列処理を取り入れており、大規模な盤面においても高い実行性能を確保している。これにより、短時間で複雑な盤面状況の分析および最適な操作の探索が可能となっている。

No.4

開発環境
使用言語：Python, C++
使用ライブラリ：OpenSiv3D
使用ソフト：Visual Studio Code, Visual Studio