

CODE VS Reborn 予選

クライアント ver.1.0 系

はじめに

このルールは体験版用に作られたものです。予選のルールとは異なる可能性がありますので予めご了承ください。

ゲームの目的

このゲームは、フィールドの一定の高さまでブロックを積み上げないようにしながら、相手のフィールドをブロックでいっぱいにするゲームです。ブロックは条件を満たすと消えます。消えたときの条件次第では相手に消えないブロックが送られ、妨害できます。うまくブロックを積み上げ消すことで、相手より長く生存することを目指しましょう。

ゲームの画面



図1. ゲーム画面のスクリーンショット

1... フィールド

フィールドは正方形のグリッドで構成される長方形です。

2...ブロック

ブロックは数値を持っており、条件を満たすと消えます。

3...お邪魔ブロック

消すことができないブロックです。

4...パック

ブロックの塊です。フィールドにはパック単位でブロックを設置します。

5...ネクストパック

次のターンのパックが表示されます。

6...デンジャーライン

このラインまでブロックが積み上がるとそのプレイヤーの負けです。

7...スキルゲージ

一定まで溜まるとスキルを発動することができます。

基本ルール

ゲームを理解するために必要なルールが記載されています。
AI実装およびシミュレーション実装などに必要な事項は、「ルールの詳細」をご覧ください。

ゲームの流れ

AIは「どのようにパックを投下するか」と「スキルを使用するか」を決めます。
パックの投下時に指定できるのは次の情報になります。

- パックの投下位置
- パックの回転数

パックを投下、またはスキルを使用してから次のパックを投下するまでの間を「ターン」と呼びます。ゲーム全体の流れは「図. ゲームの流れ」をご覧ください。

ブロック

ブロックには1～9の数値が振られています。この数値はブロックの消滅時に考慮されます。（詳細はブロックの消滅の項を参照）

お邪魔ブロック

お邪魔ブロックは消滅しないブロックです。消滅しない以外はブロックと同じ扱いです。

ブロックの消滅

ブロックは条件を満たすと消滅します。

ブロックの消滅条件は下記のようになります。

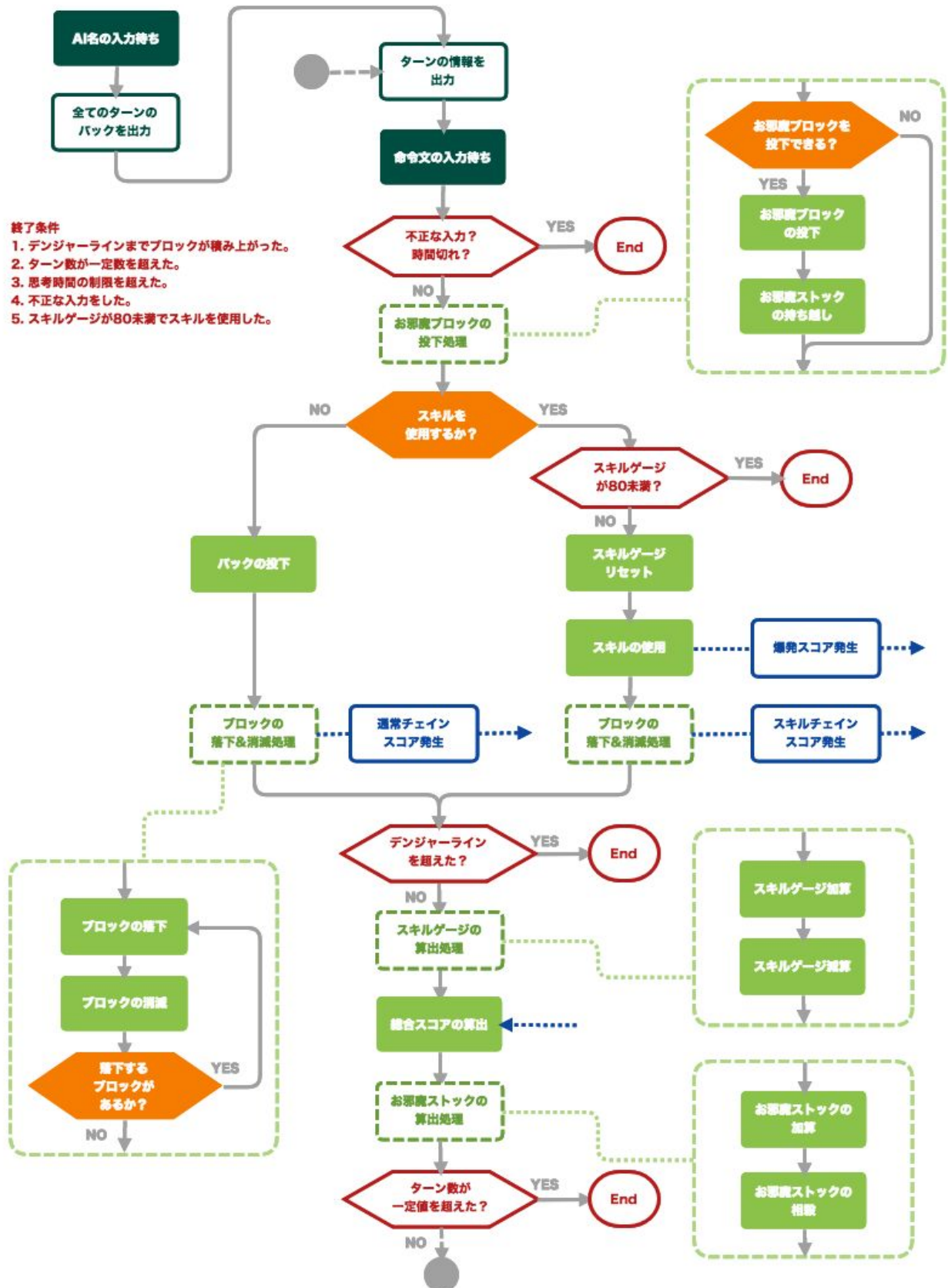
- 縦・横・斜めのいずれかの方向に隣り合う2つのブロックの数値の和が10である

条件を満たしたブロックは消滅します。またこの時、和の対象となった範囲を消滅範囲と呼びます。消滅範囲が同時に複数存在する場合は、消滅範囲に含まれる全てのブロックが消滅します。

ブロックの落下

ブロックはフィールドの底または積まれたブロックに達するまで落下します。ブロックが消滅した場合、上に乗っているブロックはすべて落下します。

図. ゲームの流れ



チェーン

1ターンのうちにブロックの落下と消滅が交互に発生します。落下と消滅の繰り返しをチェーンと呼びます。

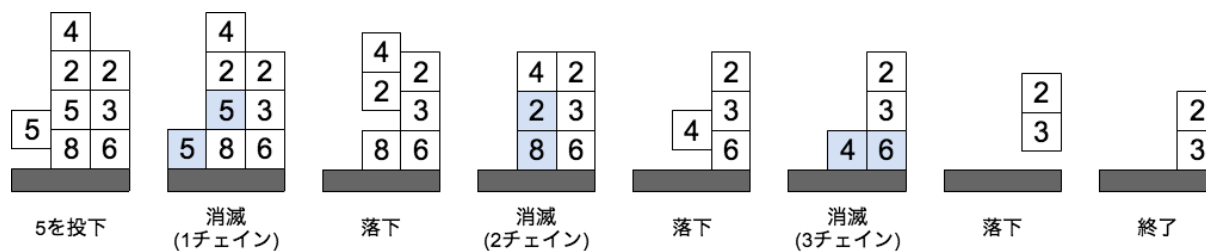


図2. 3チェーンの例（和が10の場合）

対戦相手への妨害

スコアを獲得することによって相手のフィールドにお邪魔ブロックが投下されます。また自分のフィールドに投下予定のお邪魔ブロックを減らす（相殺と呼称）こともできます。（お邪魔ブロックの投下と相殺については「ルールの詳細」で解説します）

スコア

スコアはチェーン数またはスキル使用時に消滅したブロック数などを利用した特定の式によって算出されます。（計算式については「ルールの詳細」で解説します）

スキル

スキルを使用すると、盤面上の特定のブロックが消滅し、そのブロックの周囲八方向のブロックも一緒に消滅します。スキルを使用するために、スキルゲージを一定値まで貯める必要があります。（スキルゲージの増減やスキル使用時のスコアへの影響は「ルールの詳細」で解説します）

ゲームの終了条件

下記の条件のいずれかを満たした場合、ゲームが終了します。

- ターン終了時、デンジャーラインまでブロックが積み上がった
- 思考時間の制限を超えた
- ターン数が500ターンを超えた
- 不正な入力をした

勝敗について

勝利：ターン終了時、相手がデンジャーラインまでブロックが積み上がった
相手のAIが思考時間の制限を超えた
相手のAIが不正な入力をした

敗北：ターン終了時、自分がデンジャーラインまでブロックが積み上がった
自分のAIが思考時間の制限を超えた

自分のAIが不正な入力をした

また下記の条件のとき、獲得スコアの多い方が勝利となります。

- 自分と相手が同ターンでデンジャーラインまでブロックが積み上がった
- 自分と相手のAIが同ターンで思考時間の制限を超えた
- ターン数が一定数を超えた

獲得スコアが同じ場合は、そのゲームは引き分けとなります。

また下記の条件のときは引き分けとなります。

- 自分と相手が同ターンで不正な入力した

不正な入力について

不正な入力なるのはたとえば下記のようなケースです。

- 投下位置の指定が正しくない
 - 回転数の指定が正しくない
 - スキルを使用できないタイミングでスキルを使用する
- (入力に関する詳細は「命令文の書式」で解説します)

思考時間の制限について

思考時間とは、AIが行動を決定するために計算する時間のことを言います。(思考時間の制限は「ルールの詳細」で解説します)

ルールの詳細

AIの実装およびゲームのシミュレーションする上で必要な細かな仕様が記載されています。

フィールドについて

フィールドの大きさは、横幅は10ブロック、高さは16ブロックあります。
高さ17ブロック目がデンジャーラインとなります。

ターンについて

1ゲームは最大500ターンです。

パックについて

パックは2x2の正方形の枠で、枠の中にブロックがランダムに配置されます。
1つのパックに配置されるブロックの数は3~4個です。

ブロックについて

ブロックの数値は1~9までの数が振られます。
すべてのターンを通して、1~9のブロックはそれぞれ180個与えられます。

パックの生成について

1~9の各ブロック180個（合計1,620個）を500ターン分のパックにランダムに詰めます。
3個のブロックがあるパックは380個、4個のブロックがあるパックは120個生成されます。
パックの並び順はランダムです。

消滅条件の和について

消滅条件である和の値は10です。

お邪魔ブロックについて

お邪魔ブロックは消滅しないブロックです。消滅しない以外はブロックと同じ扱いです。
チェインやスキル使用によっても消滅することはありません。「消滅条件の和+1」の数値
（つまり11）が振られたブロックとして入力で与えられます。

パックの投下

パックの投下位置と回転数を指定して、パックをフィールドに投下します。指定した投下位置と回転数が不正だった場合、ゲームは直ちに終了します。

パックの投下位置

パックの投下位置は、0～8まで指定できます。0はパックの左端がフィールドの左端に、8はパックの右端がフィールドに右端に合うようになります。「図. 投下位置のイメージ」をご参照ください。

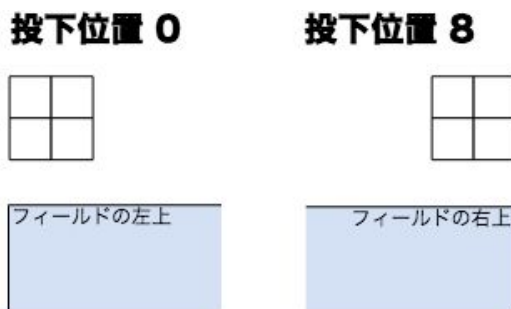


図. 投下位置のイメージ

パックの回転数

パックの回転数は、0～3の整数を指定できます。回転数1につき時計回りに90°回転します。



図. ブロックの回転数の例

チェーンによるスコアについて

ブロックを消滅させたりチェーンを繋げたりすることで、スコアを獲得します（以降、通常チェーンスコアと呼称）。スコアは下記の式で算出されます。

$$\text{通常チェーンスコア} = \sum_{i=1}^C \text{floor}(1.3^i)$$

※チェーン数 C とする

※ $\text{floor}(x)$ は x の小数点以下の切り捨てを表す

スキルについて

スキルゲージを一定値まで溜めることによってスキルが使用可能となります。

スキルゲージ

0から100までの値を取ります。1刻みになります。

$$\text{スキルチェインスコア} = \sum_{i=1}^{SC} \text{floor}(1.3^i)$$

※スキルチェイン数 SC

総合スコアについて

加点対象となるのは全部で3つあります。

- 通常チェインスコア
- スキルチェインスコア
- 爆発スコア

あるターンに発生したこれらのスコアを合算した値を総合スコアと呼称します。

お邪魔ブロックをお邪魔ストックに追加

スコアを獲得することで、お邪魔ブロックが相手にストックされます。これをお邪魔ストックと呼びます。ストックされるお邪魔ブロックの数は、「通常のパック投下」と「スキルの使用」とで異なります。

通常のパック投下

$$\text{お邪魔ブロック数} = \text{floor}(\text{通常チェインスコア}/2)$$

スキルの使用

$$\begin{aligned} \text{お邪魔ブロック数} &= \text{floor}(\text{爆発スコア}/2) \\ &+ \text{floor}(\text{スキルチェインスコア}/2) \end{aligned}$$

※ $\text{floor}(x)$ は x の小数点以下の切り捨てを表す

お邪魔ブロックの計算例

通常の3チェインの時：

通常チェインスコア

$$\begin{aligned} &\text{floor}(1.3^1) + \text{floor}(1.3^2) + \text{floor}(1.3^3) \\ &= \text{floor}(1.3) + \text{floor}(1.69) + \text{floor}(2.197) \\ &= 1 + 1 + 2 \\ &= 4 \end{aligned}$$

追加されるお邪魔ブロック

$$\begin{aligned} &\text{floor}(4 / 2) \\ &= 2 \text{個} \end{aligned}$$

スキルを使用した時（爆発したブロック数5個、そこから5チェイン発生）：

爆発スコア

$$\begin{aligned} &\text{floor}(25 \cdot 2^{5/12}) \\ &= \text{floor}(33.370...) \\ &= 33 \end{aligned}$$

スキルチェインスコア

$$\begin{aligned}
 & \text{floor}(1.3^1) + \text{floor}(1.3^2) + \text{floor}(1.3^3) + \text{floor}(1.3^4) + \text{floor}(1.3^5) \\
 &= 1 + 1 + 2 + 2 + 3 \\
 &= 9
 \end{aligned}$$

追加されるお邪魔ブロック

$$\begin{aligned}
 & \text{floor}(33 / 2) + \text{floor}(9 / 2) \\
 &= 16 + 4 \\
 &= 20\text{個}
 \end{aligned}$$

お邪魔ブロックの投下

お邪魔ブロックの投下は下記の流れで行われます。

1. お邪魔ブロックの投下
2. お邪魔ストックの持ち越し
3. (パックの投下)
4. (ブロックの消滅&落下処理)
5. スキルゲージの加算
6. スキルゲージの減算
7. お邪魔ブロックをお邪魔ストックに追加
8. お邪魔ストックの相殺
9. (ターン終了)
10. (システムからの入力：次のターン)
11. お邪魔ブロックの投下

お邪魔ブロックの相殺

両者のお邪魔ブロック数がどちらも0でない場合、少なくとも片方が0になるまで両方からお邪魔ブロックを1つずつ減らします。これをお邪魔ブロックの相殺と呼びます。

例1.

自分のお邪魔ストック10個、相手のお邪魔ストックが5個

- ⇒ 自分のお邪魔ストック9個、相手のお邪魔ストック4個
- ⇒ 自分のお邪魔ストック8個、相手のお邪魔ストック3個
- ⇒ 自分のお邪魔ストック7個、相手のお邪魔ストック2個
- ⇒ 自分のお邪魔ストック6個、相手のお邪魔ストック1個
- ⇒ 自分のお邪魔ストック5個、相手のお邪魔ストック0個

最終的な次のターン開始時のお邪魔ブロックの数：

自分のお邪魔ストック5個、相手のお邪魔ストック0個

例2.

自分のお邪魔ストック8個、相手のお邪魔ストックが10個

- ⇒ 自分のお邪魔ストック0個、相手のお邪魔ストック2個

例3.

自分のお邪魔ストック0個、相手のお邪魔ストックが2個

- ⇒ 自分のお邪魔ストック0個、相手のお邪魔ストック2個

お邪魔ブロックの投下

ターン開始時にお邪魔ストックがフィールドの横幅以上に溜まった場合、横一列にお邪魔ブロックが投下されます。お邪魔ストックの数は投下された分、減少します。

例：横幅が10マスの場合、お邪魔ストックに10以上のとき、お邪魔ブロックが10個横一列に投下され、お邪魔ストックの数が10減少します。

お邪魔ストックの持ち越し

投下しきれずに残ったお邪魔ストックは次のターンに持ち越されます。

思考時間の制限について

思考時間は**1試合3分**までと制限されています。1ターンの思考時間の制限はありません。さらに予選では、1ターンにつき思考時間は20秒と制限されています。ただし、クライアントを使用してローカルで対戦を行ったときは、1ターンの思考時間の制限はないものとして対戦が行われるのでご注意ください。オンラインで対戦を行った場合は、ターンの時間制限がある状態で対戦が行われます。

AIの出力

AIは、決められた書式に従って標準出力にテキストを出力することでゲームに対して行動を指定してください。標準出力に出力するテキストを命令文と呼称します。

また命令文はそのターンのシステムの入力を受け取った後に出力してください。

よくある質問

Q. 標準出力に出力したのに読み取ってくれないのはなぜ？

A. 言語仕様に従った標準出力のフラッシュを行ってください

AI名について

システムはまずAI名を受け付けます。AI名となる文字列を標準出力に出力してください。またAI名には「V:*?"<>|」のいずれかの文字を含めることはできません。

毎ターンの命令文の出力の際は不要です。

AI名の例

```
sample_ai
```

命令文の書式

命令文は「パックの投下」と「スキルの使用」で異なります。

パックの投下

命令文は下記の形で出力してください。

```
[投下位置] [パックの回転数]
```

例：投下位置2、パックの回転数0の場合

2 0

例：不正なケース

a 100

スキルの使用

命令文は以下の形で出力してください。

S

システムからの入力

システムからの入力は「ゲーム開始時」と「ターン開始時」に行われます。したがって、1ターン目は「ゲーム開始時の入力」と「ターンごとの入力」が連続して送られます。また、以降の説明は下記を前提とします。

ゲーム開始時の入力

ゲーム開始時の入力は下記の形で与えられます。

[1ターン目に投下されるパック情報]
[2ターン目に投下されるパック情報]
...
[500ターン目に投下されるパック情報]

パック情報

パック情報は2×2行列で1行目が最も天井に近い行、2行目が最も底に近い行になります。各行には、その行のブロックのもつ数値が半角スペース区切りで与えられます。0は空白を意味します。それぞれのパック情報の最終行にはENDと書かれた行が与えられます。

パック情報の例

2 0
8 1
END

ターンごとの入力

ターンごとの入力は下記の形で与えられます。

[現在のターン数]
[自分の残りの思考時間。単位はミリ秒]
[自分のお邪魔ストック]
[自分のスキルゲージ]
[自分のスコア]
[前のターン終了時の自分のフィールド情報]
[相手の残りの思考時間。単位はミリ秒]

[相手のお邪魔ストック]
[相手のスキルゲージ]
[相手のスコア]
[前のターン終了時の相手のフィールド情報]

現在のターン数

与えられる入力のターン数です。1ターン目が0、最終ターンは499です。

自分のお邪魔ストック、相手のお邪魔ストック

お邪魔ブロック投下前のお邪魔ストック数が入ります。お邪魔ストックが10以上の場合、このターンの最初に横一列にお邪魔ブロックが投下されます。

自分のスキルゲージ、相手のスキルゲージ

前のターン終了時のスキルゲージの値が入ります。

自分のスコア、相手のスコア

前のターン終了時のスコアが入ります。

フィールド情報

フィールド情報は16×10行列で1行目が最も天井に近い行、16行目が最も底に近い行になります。各行には、その行のブロックのもつ数値が半角スペース区切りで与えられます。0は空白を意味します。最終行にはENDと書かれた行が与えられます。

フィールド情報の例（16 × 10 のフィールド）

```
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 5 0 0 0 0 0
0 0 0 0 1 0 1 0 3 0
END
```

入力例

ゲームの例として、実際の入力を示します。

また、「#」から改行までの文字列は、わかりやすさのためのコメントです。実際の入力にはありません。

```
9 8
9 2
```

[illegible]

```
8
1
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
9 0 0 0 0 0 0 0 0 0
9 0 0 0 0 0 0 0 0 0
```

END

179690

0

8

1

```
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
9 0 0 0 0 0 0 0 0 0
9 0 0 0 0 0 0 0 0 0
```

END

2ターン目ここまで

#～中略～

499

500ターン目

179999

0

8

1

```
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
```



```

0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
2 0 0 0 0 0 0 0 0 0
9 7 0 0 0 0 0 0 0 0
9 8 0 0 0 0 0 0 0 0

```

END

179594

0

8

1

```

0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
8 0 0 0 0 0 0 0 0 0
7 0 0 0 0 0 0 0 0 0
9 0 0 0 0 0 0 0 0 0
9 2 0 0 0 0 0 0 0 0

```

END

500ターン目ここまで

以上の入力では、以下の3つのパックがこの順番で与えられていることになります。

1ターン目	2ターン目	500ターン目												
<table><tr><td>9</td><td>8</td></tr><tr><td>9</td><td>2</td></tr></table>	9	8	9	2	<table><tr><td>2</td><td>7</td></tr><tr><td></td><td>8</td></tr></table>	2	7		8	<table><tr><td>9</td><td>5</td></tr><tr><td></td><td>3</td></tr></table>	9	5		3
9	8													
9	2													
2	7													
	8													
9	5													
	3													

図7. 入力例のパック

更新履歴

2019/04/16：記述のミスを修正

変更点

- パックの生成についての記述ミスを修正
- システムからの入力 > 自分のお邪魔ストック、相手のお邪魔ストック の記述ミスを修正

2019/04/15：予選ルール公開

変更点

- システムからの入力に入力に前のターンのスコアを追加

2019/03/26：体験版ルール公開