

産学連携と数理・暗号分野連携による
カードベース暗号の深化と新境地Ⅲ

カードベース暗号を用いた ヒット＆ブローの遊び方

発表者：池田 昇太（茨城大学）

共同研究者：品川 和雅（筑波大学/産業技術総合研究所）

本研究の概要

カードベース暗号を用いたヒット＆ブローの遊び方

*池田昇太, 品川和雅, CSEC研究会 2024

How to Play Mastermind without Game Master.

*Shota Ikeda, Kazumasa Shinagawa, TAMC 2025

- ヒット＆ブローは出題者の秘密の数字(コード)を解答者が当てるゲーム
- 提案プロトコルを用いれば、出題者不在でもヒット＆ブローが遊べる

目次

- 第1部:はじめに
 - ヒット&ブローとは
 - 本研究の貢献
 - カードと符号化とシャッフル
- 第2部:提案プロトコルと遊び方
 - 提案プロトコルの手順
 - 遊び方の例
- まとめ

第1部 はじめに

ヒット＆ブロー

- 出題者の持つ秘密のコード(数字or色)を解答者が推測するゲーム

おもちゃの例



ゲーム画面の例(PC・スマホゲーム)

目標コード (出題者の秘密)

? ? ? ?

遊び方

- 4桁の数字を推測します (数字は0~9、重複なし)
- 数字を入力して推測します
- Hit と Blow の数が表示されます
- 🔄 Hit が4になるまで繰り返します

推測を入力

決定

試行	推測 (プレイヤーの入力)	結果 (Hit / Blow)	結果の見方
1	0 2 1 5	1 Hit 1 Blow	● ●
2	3 4 6 8	0 Hit 2 Blow	● ●
3	4 7 2 1	2 Hit 1 Blow	● ● ●
4	4 6 2 7	3 Hit 0 Blow	● ● ●
5	4 6 2 1	4 Hit 0 Blow	● ● ● ●
6		-	
7		-	
8		-	

● Hit (ヒット)
= 数字も位置も正しい

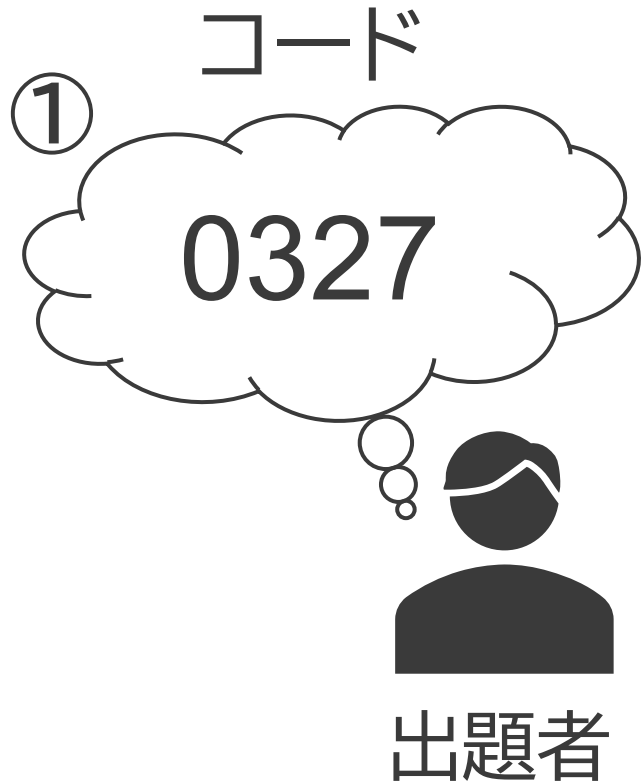
● Blow (ブロー)
= 数字は正しいが位置が違う

4 Hit !
= すべての位置と数字が一致!

※画像はイメージ

ヒット&ブローの流れ

① 出題者がコードを設定



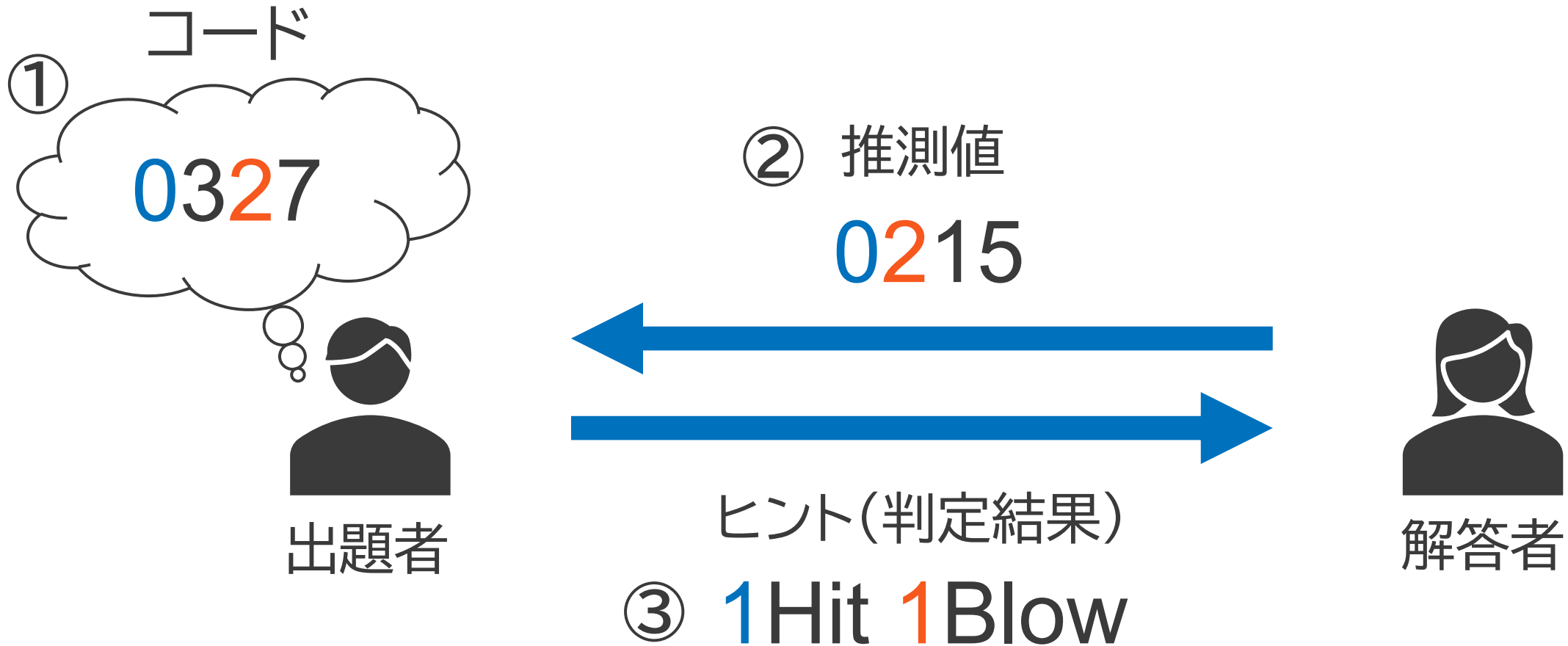
ヒット＆ブローの流れ

② 解答者が推測値を出題者に示す



ヒット＆ブローの流れ

③ 出題者はコードと推測値を比較し、ヒントを返す



ヒット&ブローのヒント

コードと推測値を比較して

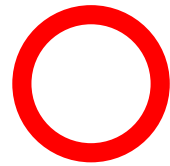
- 同じ数字が同じ桁にある場合
→ **"Hit"**.
- 同じ数字が異なる桁にある場合
→ **"Blow"**.

ex) コード : 0 3 2 7
 推測値 : 0 2 1 5
 → 1Hit 1Blow

各桁の数字の重複

- コード・推測値で各桁の数字の重複を許さないルールが一般的
 - ゲームバランスが良い
 - 多くのゲームwebサイトがこのルールを採用

例)



392



393

- 一部ゲームサイトでは、重複を許すルールが採用
- 本研究で重複を許さないルールを採用

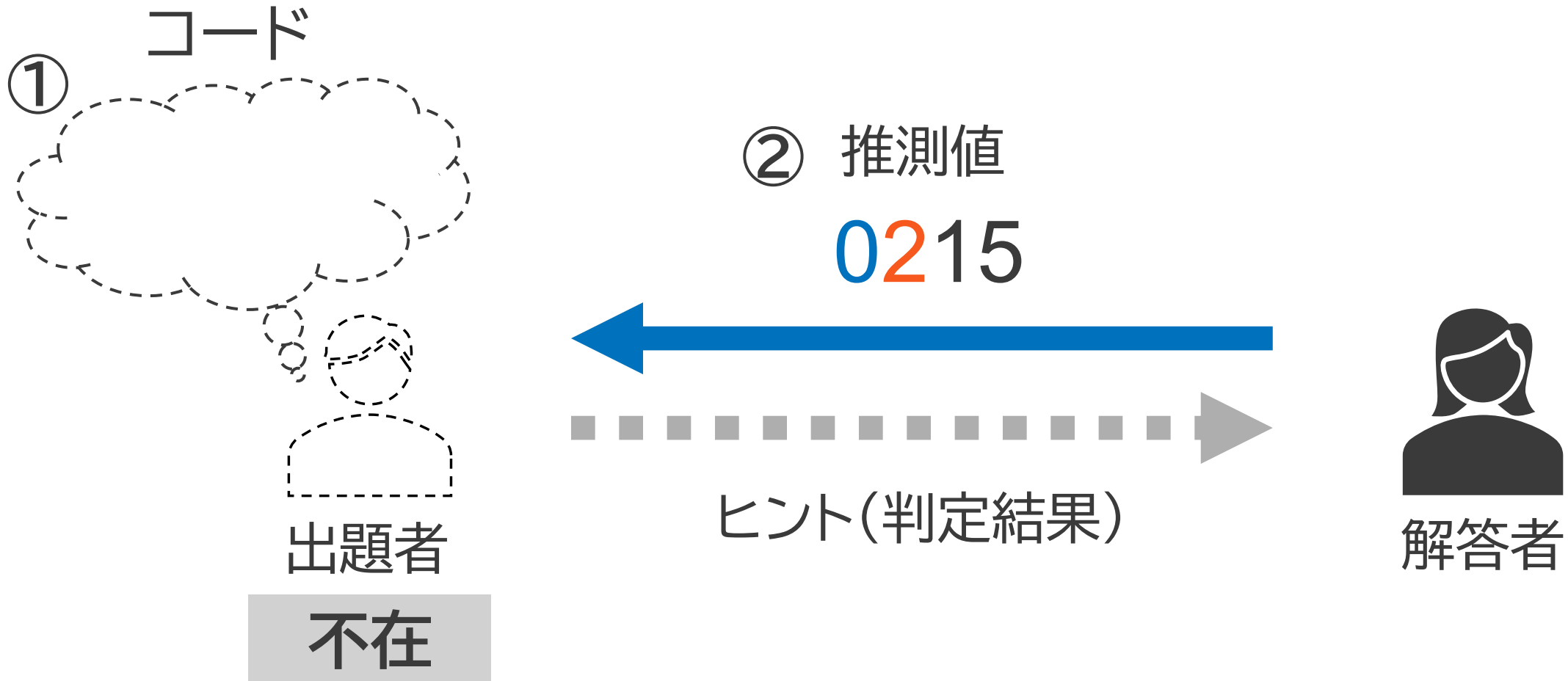
ヒット＆ブローの課題

- 出題者がウソのヒントを返す場合がある



ヒット&ブローの課題

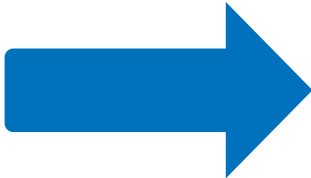
- 1人(出題者不在)では遊ぶことができない



本研究の貢献

2種のプロトコルを提案

① フィードバックプロトコル

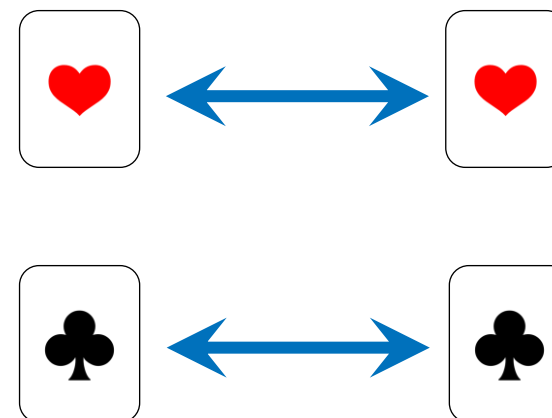
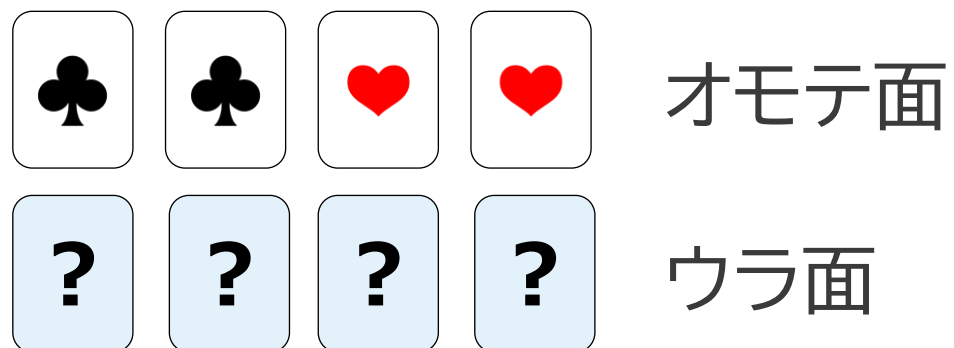
コード, 推測値:   ...   **Hit** の数 & **Blow** の数

② コード生成プロトコル

複数のコードをランダムに生成

ウソのヒントが返されない & 1人(解答者のみ)で遊べる

カード

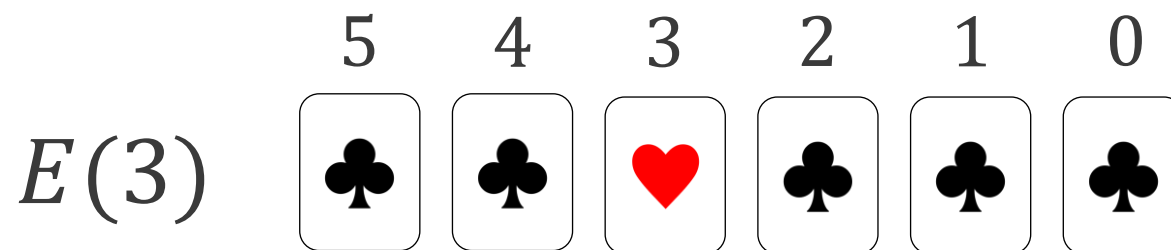


2色カード

同じ色のカードは見分けがつかない

整数符号化

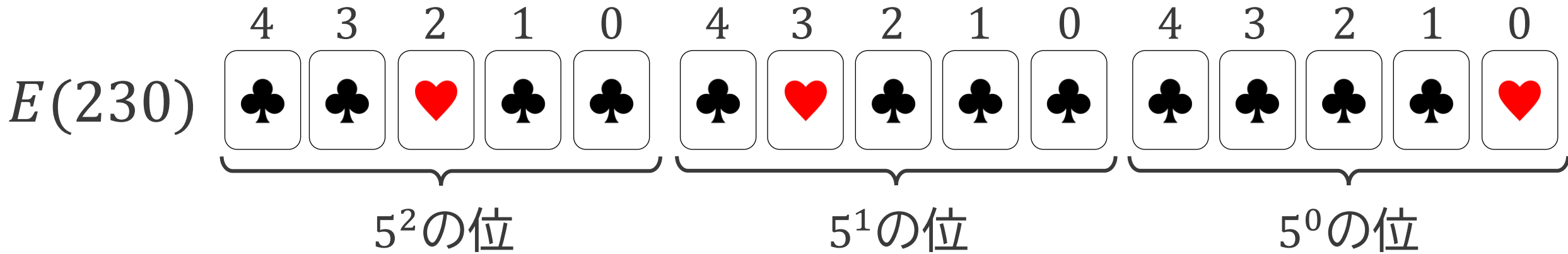
例: $E(3)$ の整数符号化



- 1枚の  と $k - 1$ 枚の  のカードで整数 $x \in \{0, 1, \dots, k - 1\}$ を表現
- x 番目に  を置き、それ以外の全てに  を置く符号化

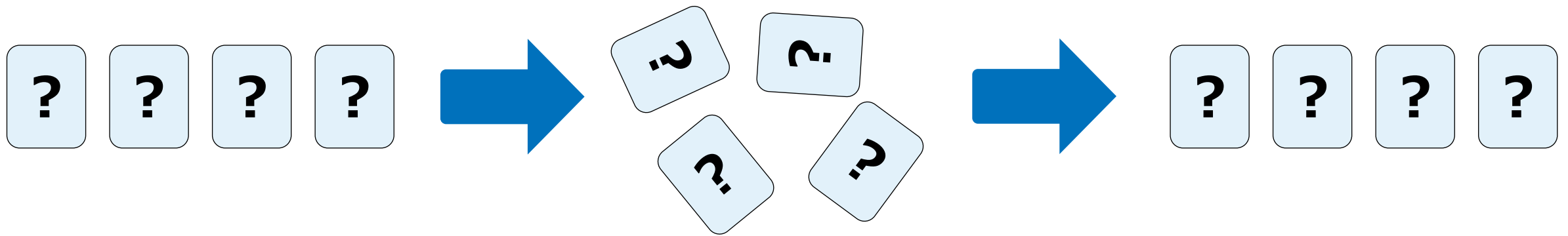
N 進 M 桁符号化

例: $E(230)$ の 5 進 3 桁符号化



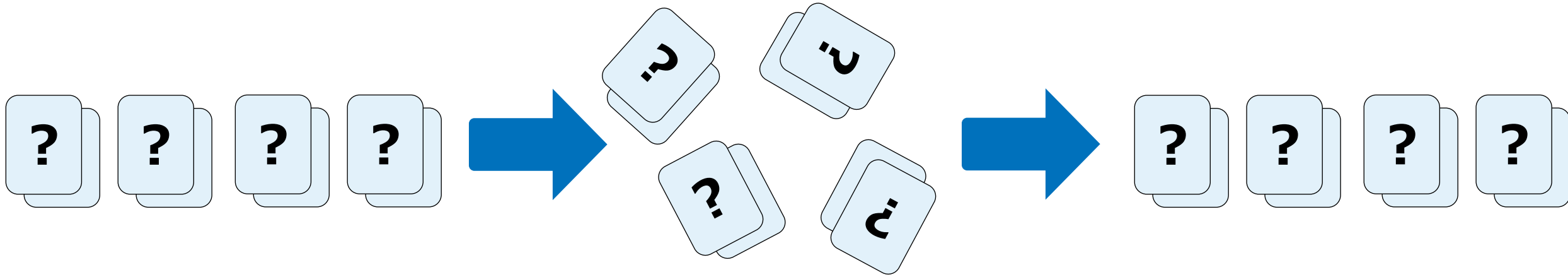
- それぞれの位で整数符号化を行う整数符号化の拡張
- 整数符号化より少ないカード枚数で大きな整数を表現可能

完全シャッフル(CS)



- カード列をぐちゃぐちゃにするシャッフル
- 麻雀の洗牌という操作に近いシャッフル

パイルスクランブルシャッフル(PSS)[ISM15]



- 同じ枚数からなるカード束を作る
- カード束をぐちゃぐちゃにかき混ぜるシャッフル
- スリーブやバンドを用いれば簡単に実行できる



第2部

提案プロトコルと遊び方

フィードバックプロトコルの比較

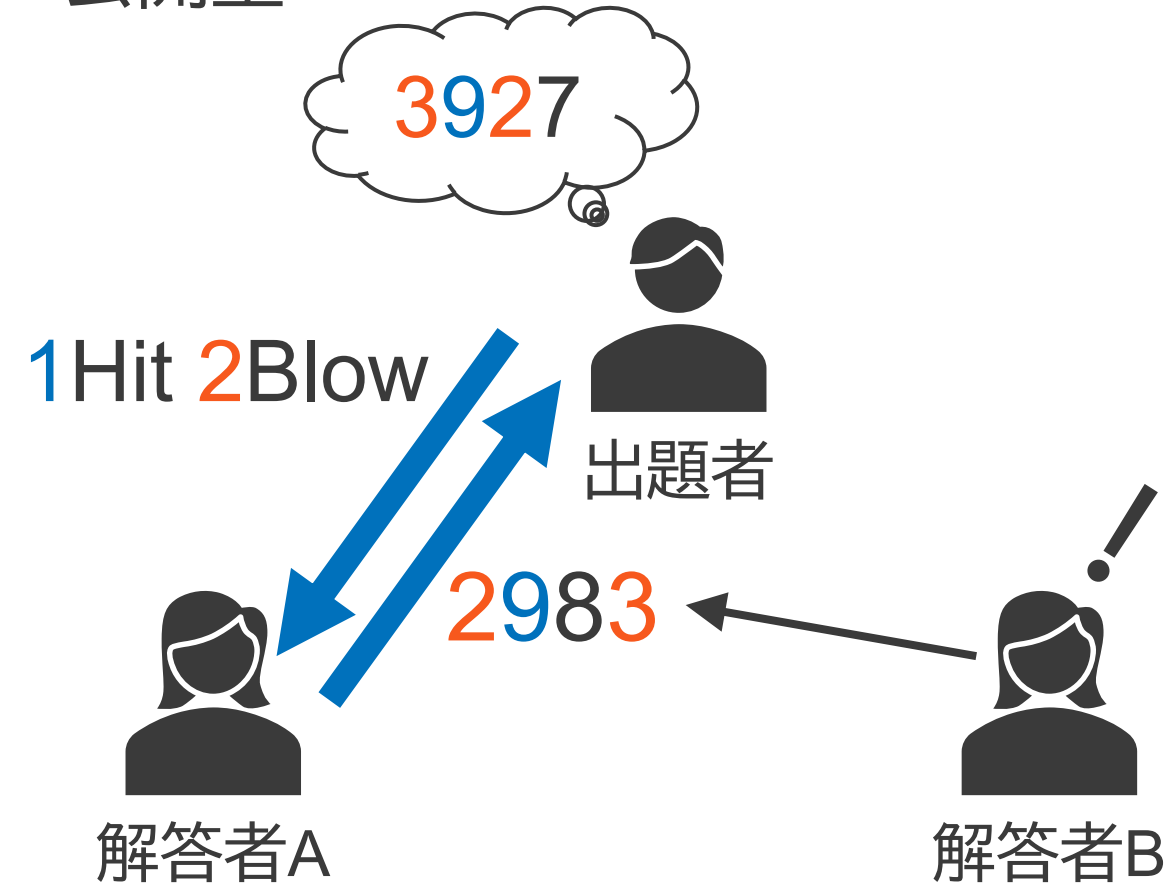
プロトコル	カード枚数(枚)	シャッフル(回)	推測値の入力
推測値公開	NM	2CS	公開値
推測値秘匿	$2NM$	2PSS	カード列
推測値公開＋コード保存	$2NM$	2CS+1PSS	公開値
推測値秘匿＋コード保存	$3NM$	3PSS	カード列

※ N 進 M 桁符号化を使用、CS:完全シャッフル、PSS:パイルスクランブルシャッフル

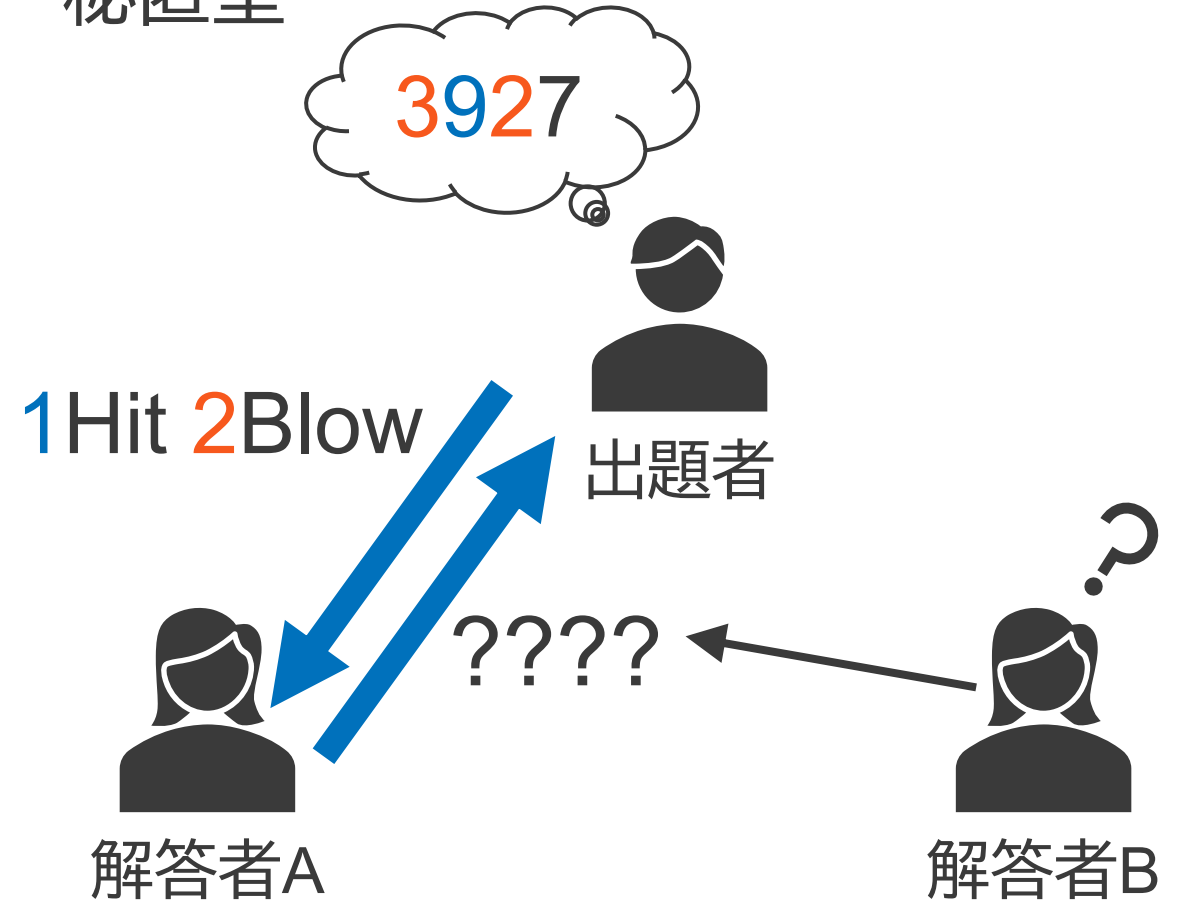
コード, 推測値:   Hit の数 & Blow の数

推測値公開型と推測値秘匿型の違い

公開型



秘匿型



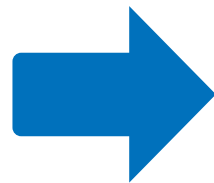
プロトコルの概要

推測値公開型フィードバックプロトコル

コード:   ... 

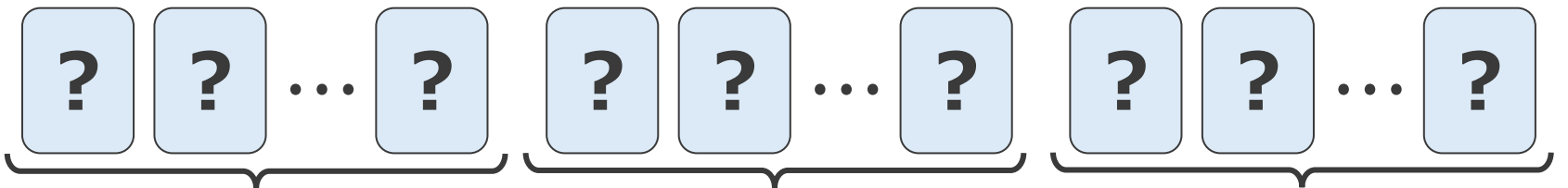
推測値: 公開値

シャッフル: **2CS**



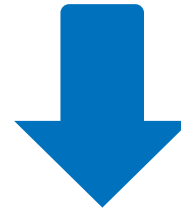
Hit の数 & **Blow** の数

プロトコルの実行手順の例

コード: $E(907)$ 
100の位 10の位 1の位

推測値: 910

公開値

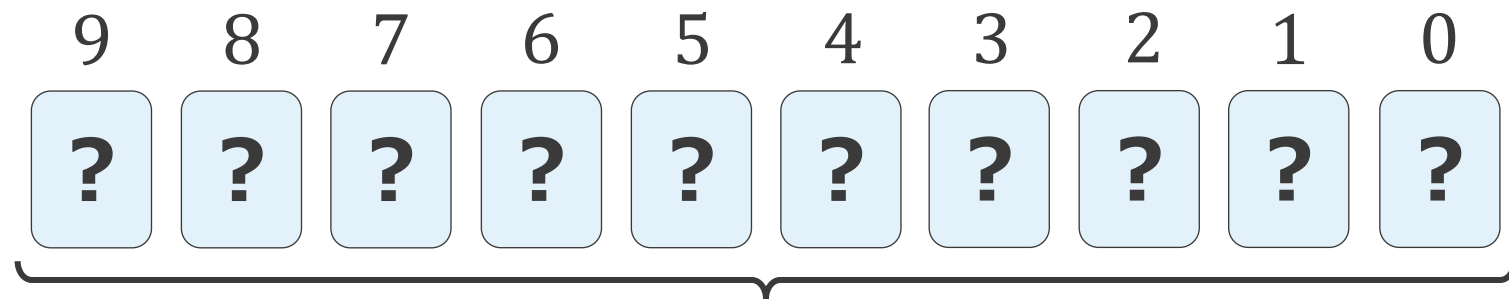


フィードバックプロトコル
(推測値公開型)

1 Hit 1 Blow

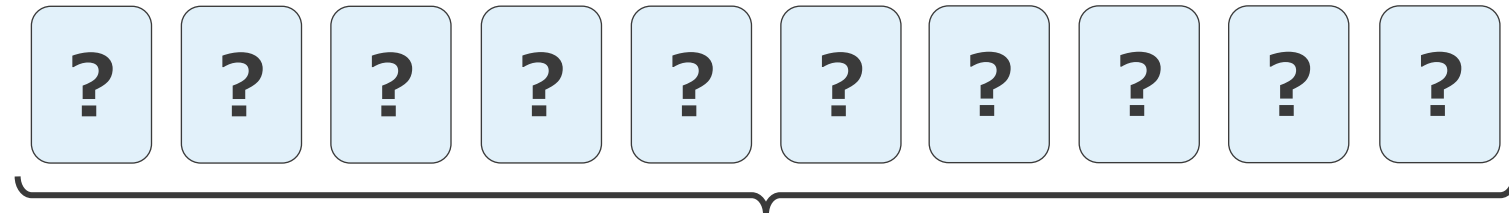
実行手順(1/5)

1. コードが以下のように与えられている

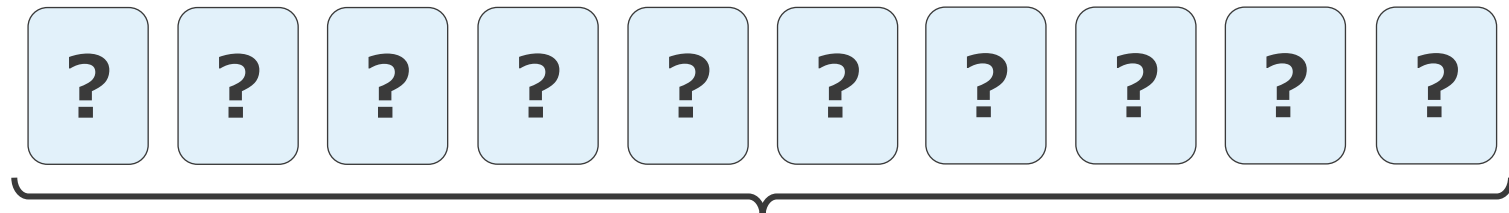


100の位

コード:



10の位

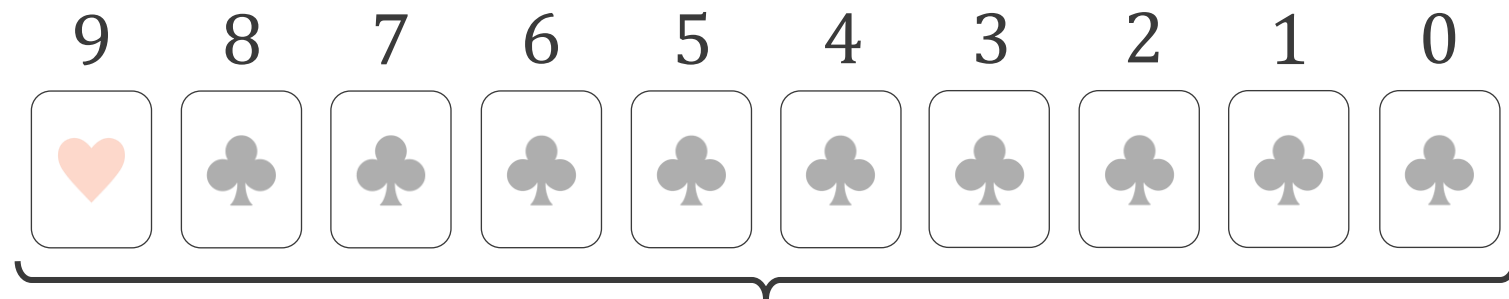


1の位

コード: 907
推測値: 910

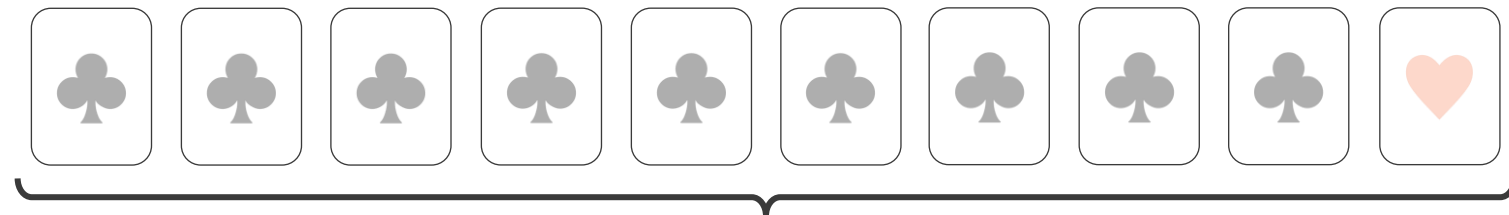
実行手順(1/5)

1. コードが以下のように与えられている

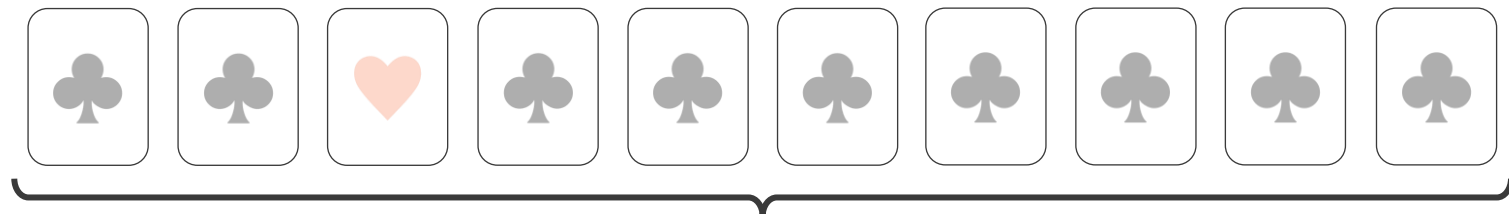


100の位

コード:



10の位

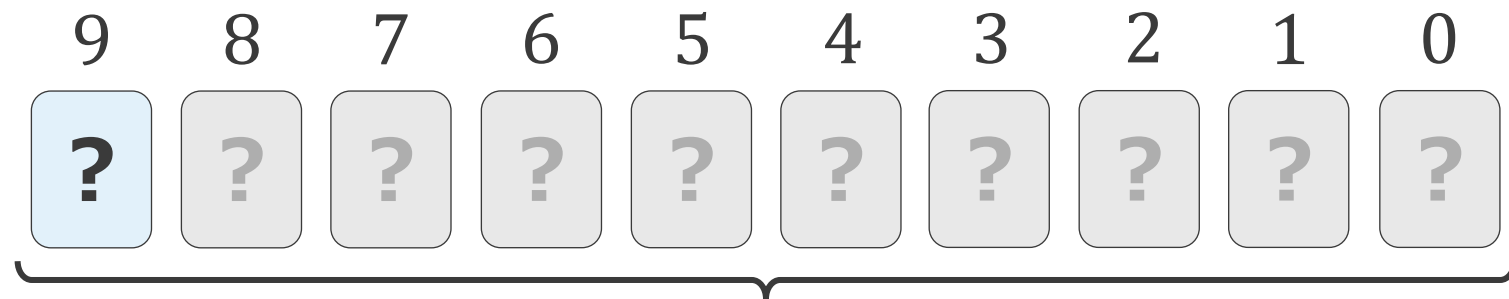


1の位

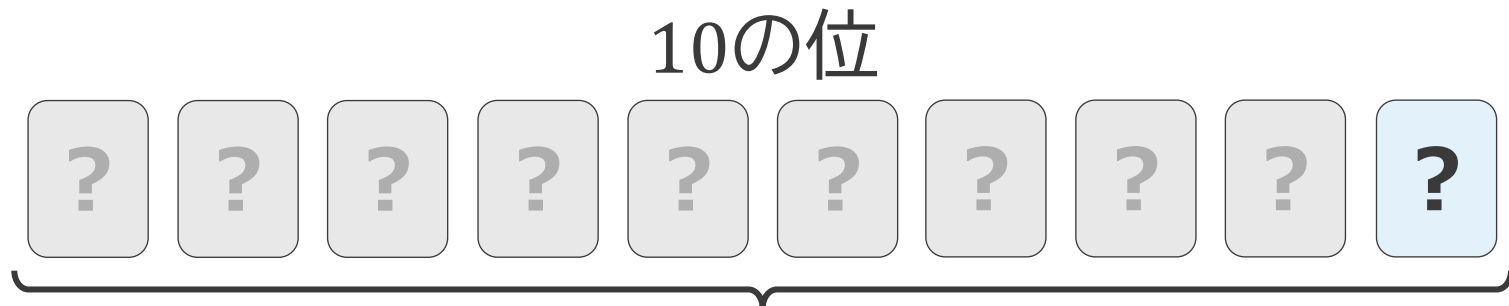
コード: 907
推測値: 910

実行手順(2/5)

2. 解答者は、各桁で推測値に対応するカードを1枚選択



コード:

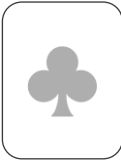
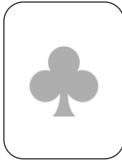
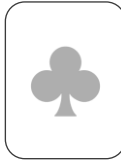
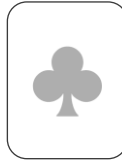
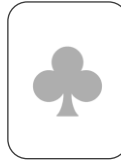
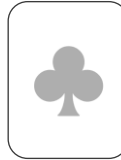
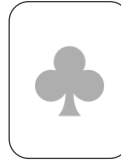
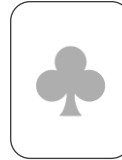
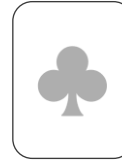
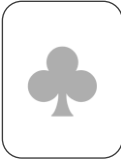
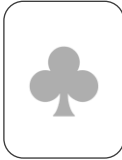
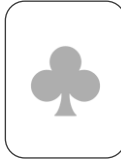
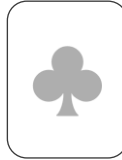
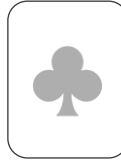
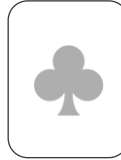
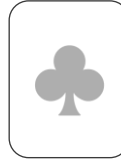
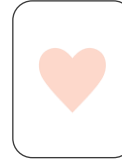
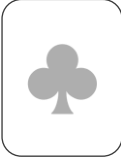
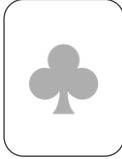
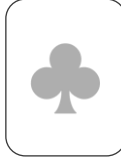
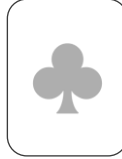


コード: 907
推測値: 910

実行手順(2/5)

2. 解答者は、各桁で推測値に対応するカードを1枚選択

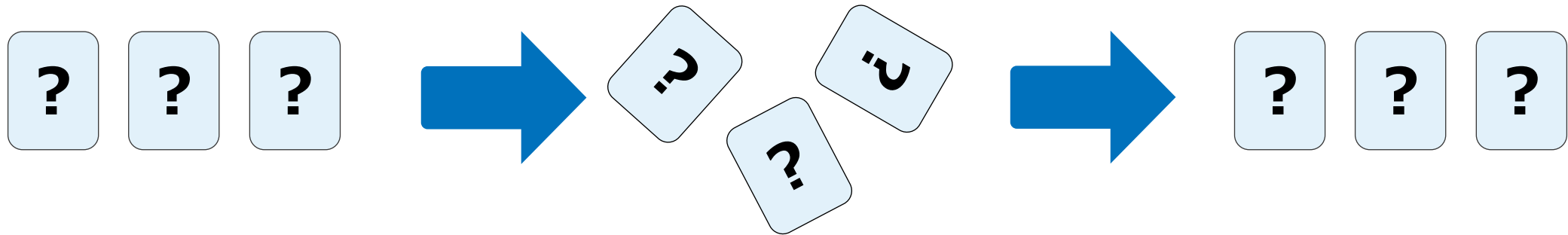
コード:

9	8	7	6	5	4	3	2	1	0
									
100の位									
									
10の位									
									
1の位									

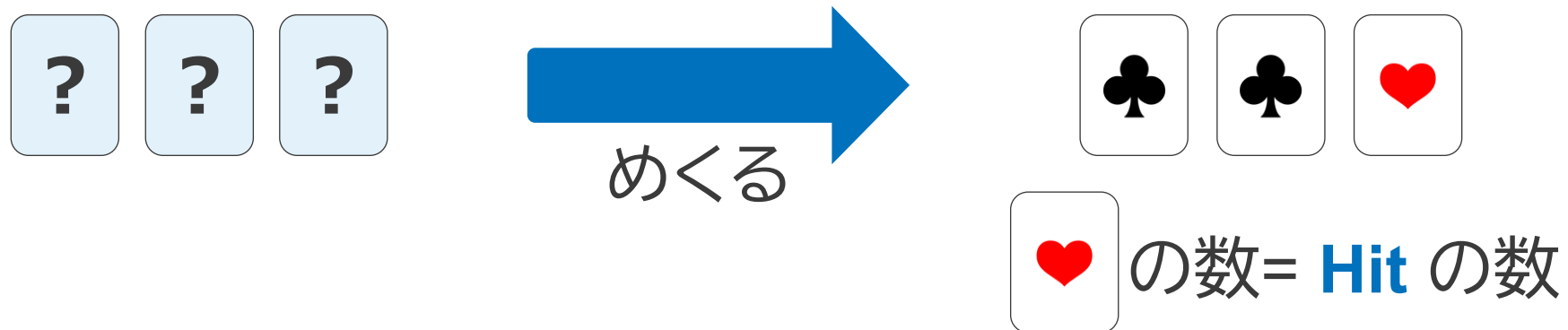
コード: 907
推測値: 910

実行手順(3/5)

3. 選んだカードに対してシャッフル(CS)を適用する

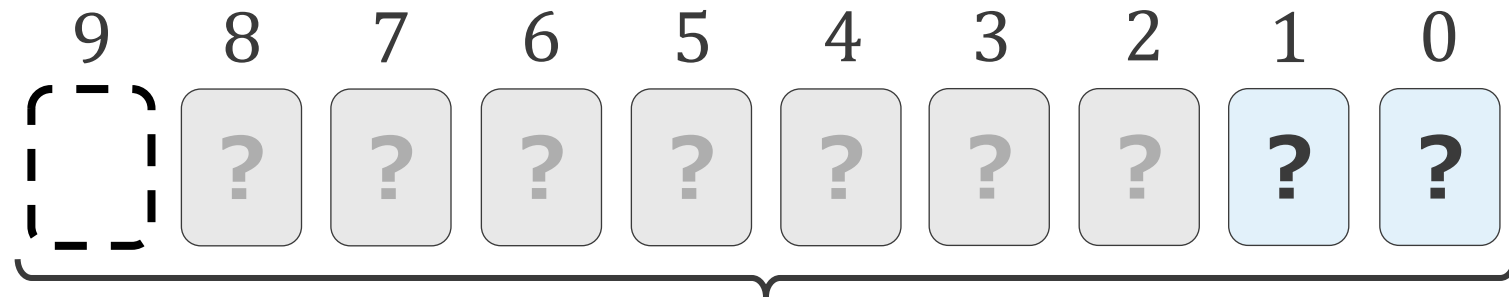


4. シャッフル後、すべてのカードをめくる

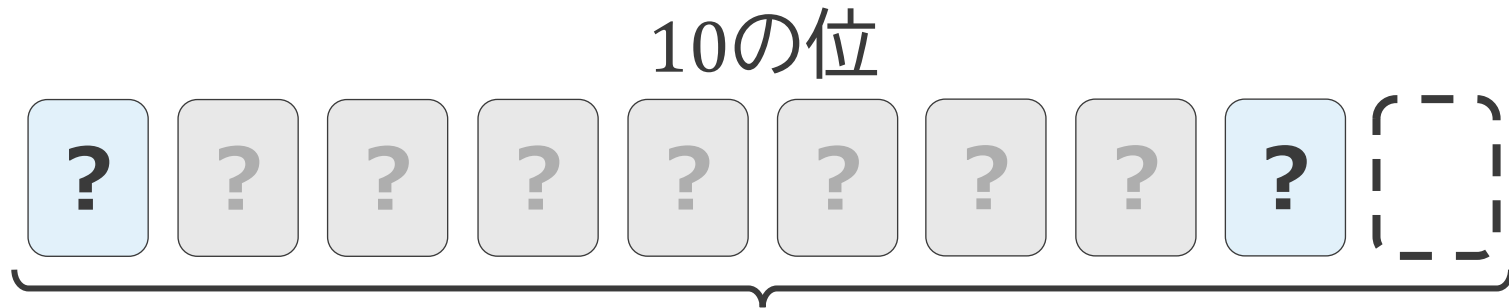
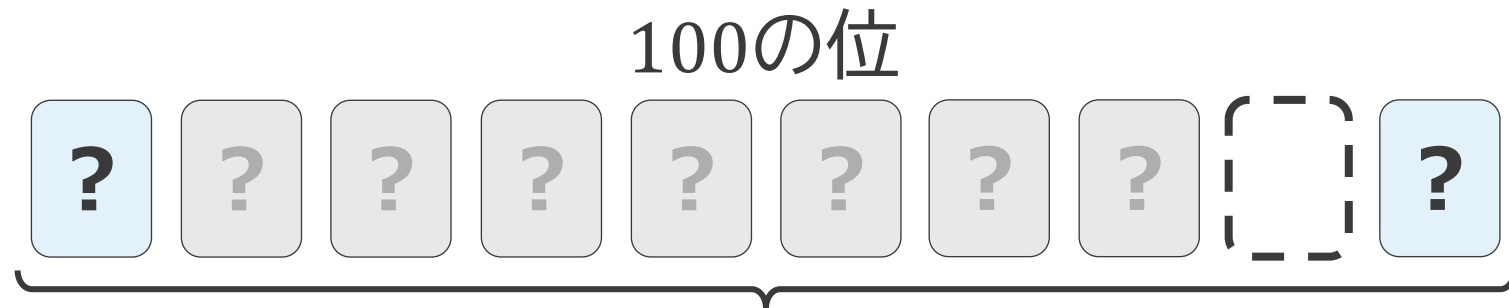


実行手順(4/5)

5. ステップ2で選択したカードと同じ数字で異なる桁のカードを選ぶ



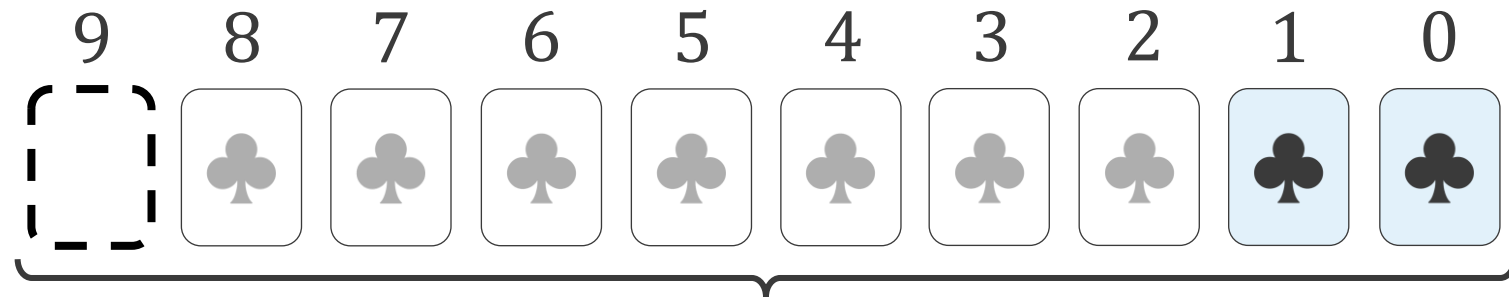
コード:



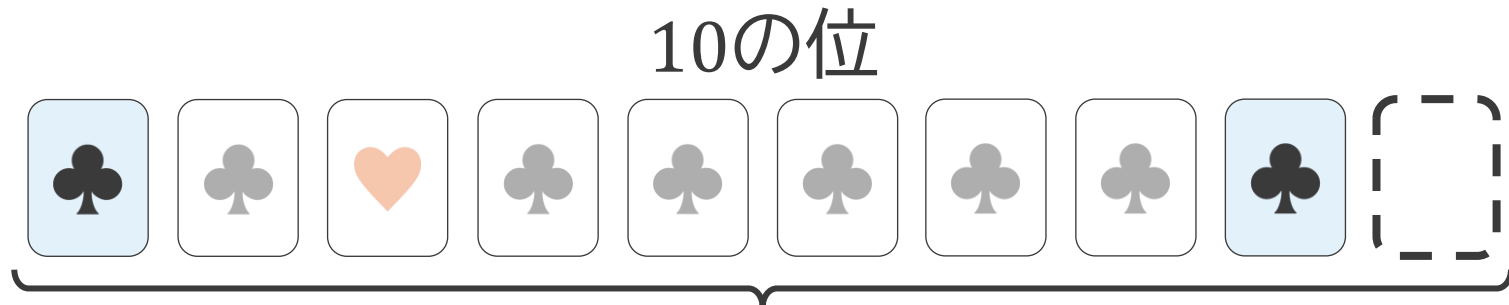
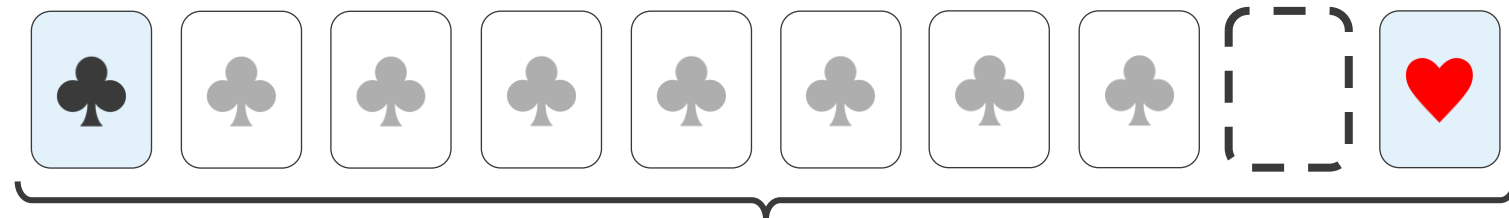
コード: 907
推測値: 910

実行手順(4/5)

5. ステップ2で選択したカードと同じ数字で異なる桁のカードを選ぶ



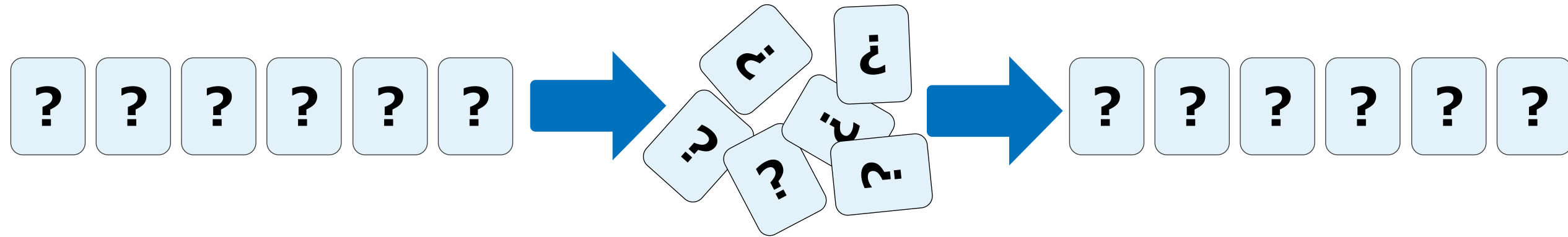
コード:



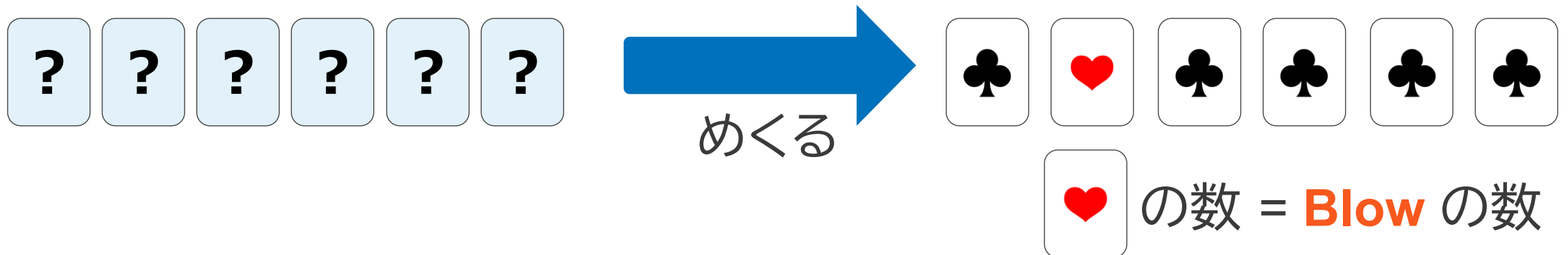
コード: 907
推測値: 910

実行手順(5/5)

6. 選んだカードに対してシャッフル(CS)を適用する



7. シャッフル後、すべてのカードをめくる



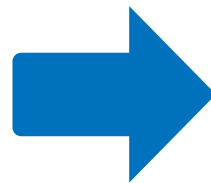
プロトコルの概要

推測値秘匿型フィードバックプロトコル

コード:   ... 

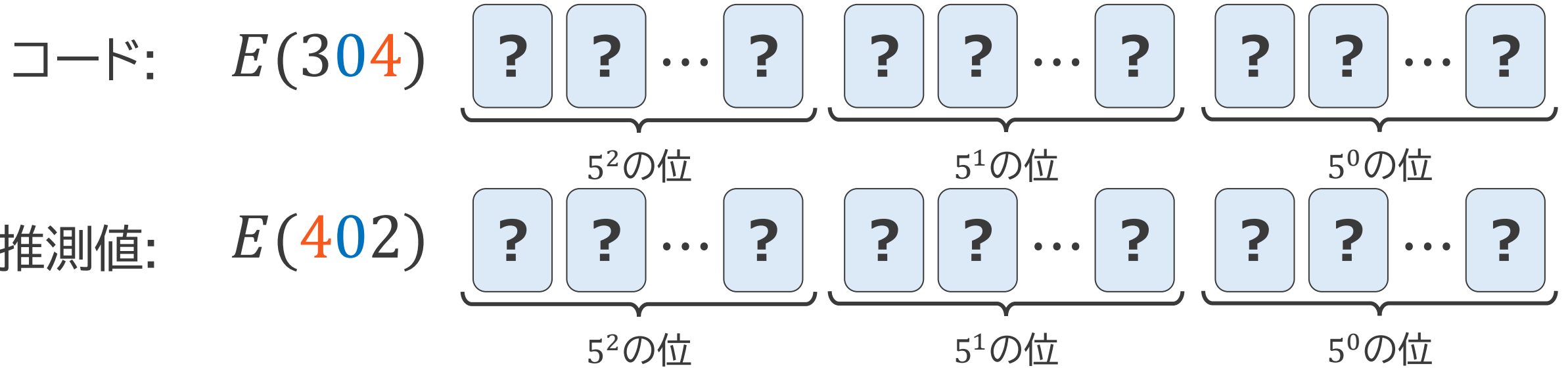
推測値:   ... 

シャッフル: **2PSS**



Hit の数 & **Blow** の数

プロトコルの実行手順の例

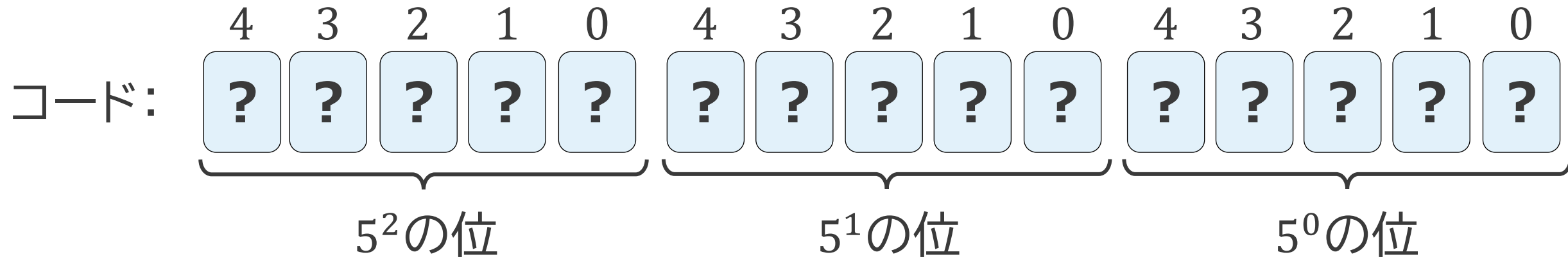


フィードバックプロトコル
(推測値秘匿型)

1 Hit 1 Blow

実行手順(1/5)

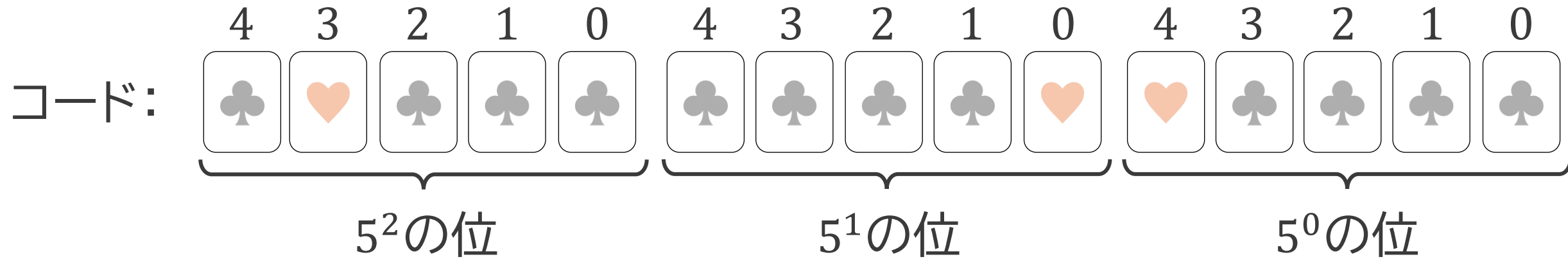
1. コードが以下のように与えられている



コード: 304
推測値: 402

実行手順(1/5)

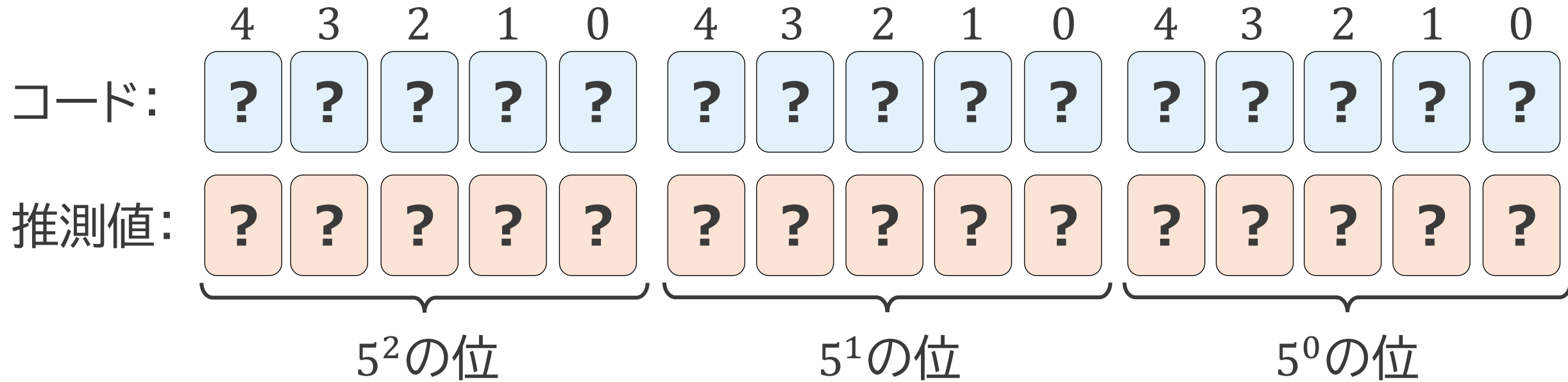
1. コードが以下のように与えられている



コード: 304
推測値: 402

実行手順(2/5)

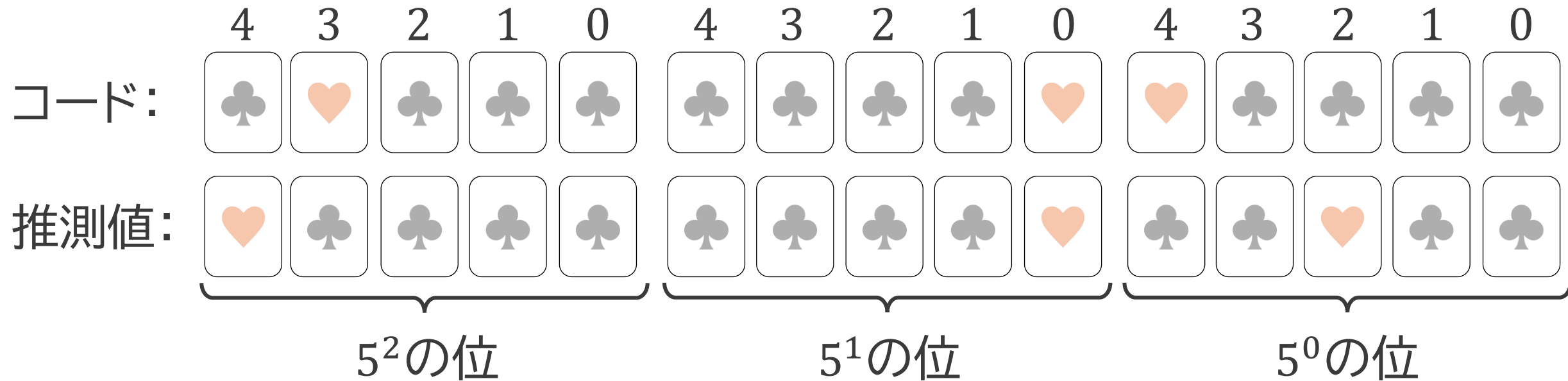
2. 解答者が推測値をカード列で与える



コード: 304
推測値: 402

実行手順(2/5)

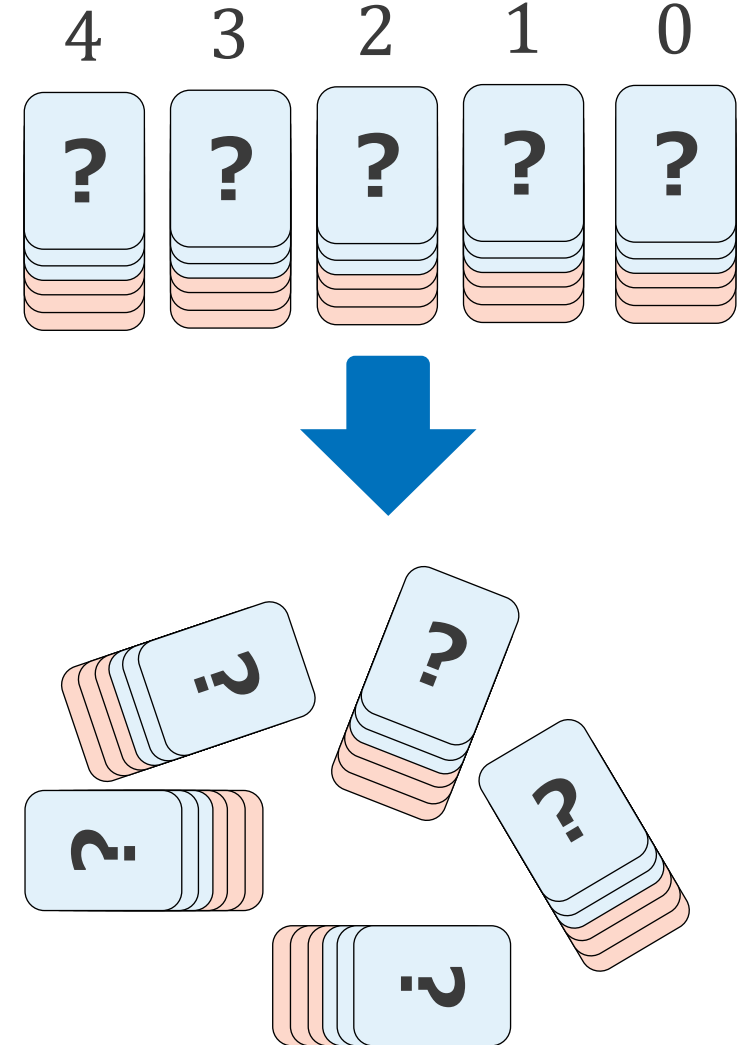
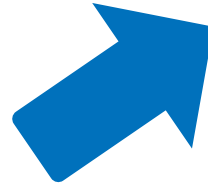
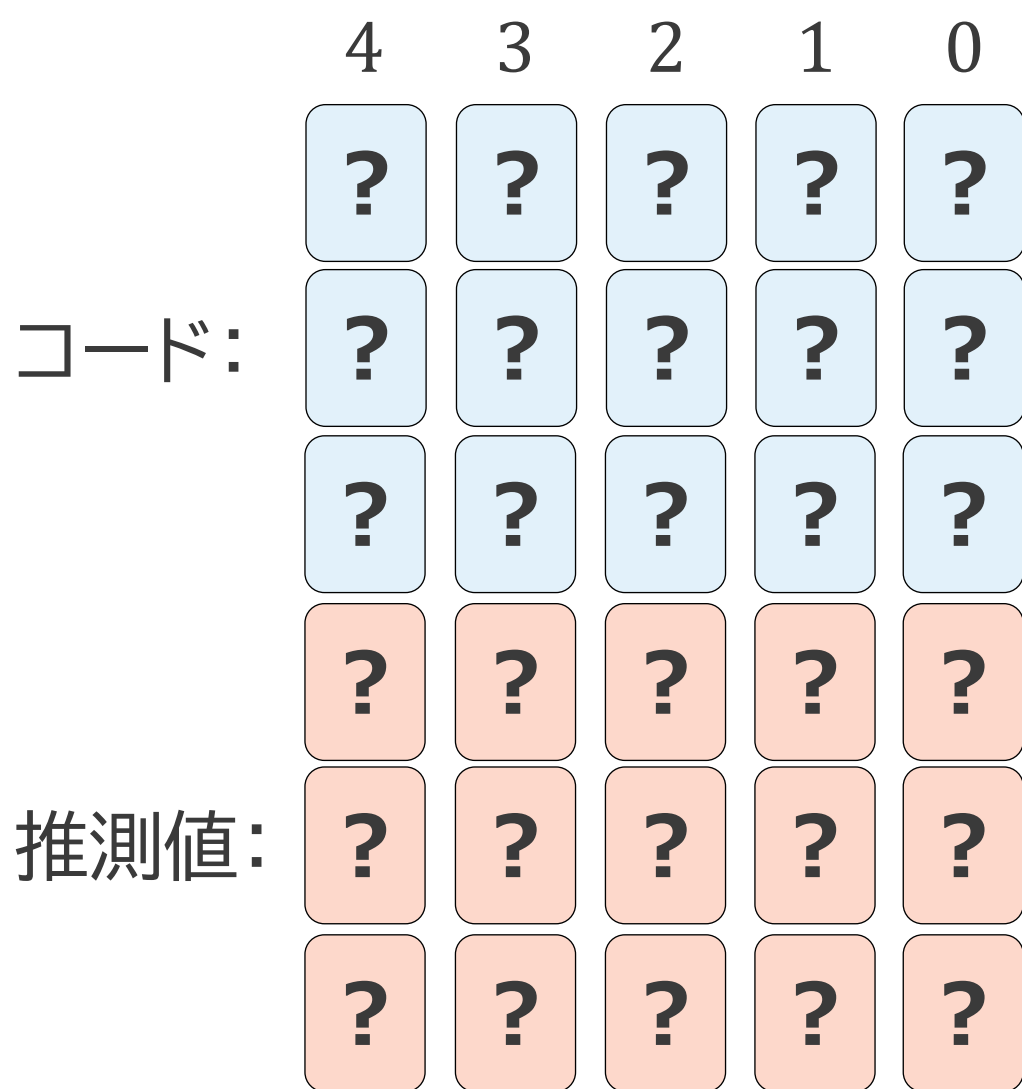
2. 解答者が推測値をカード列で与える



コード: 304
推測値: 402

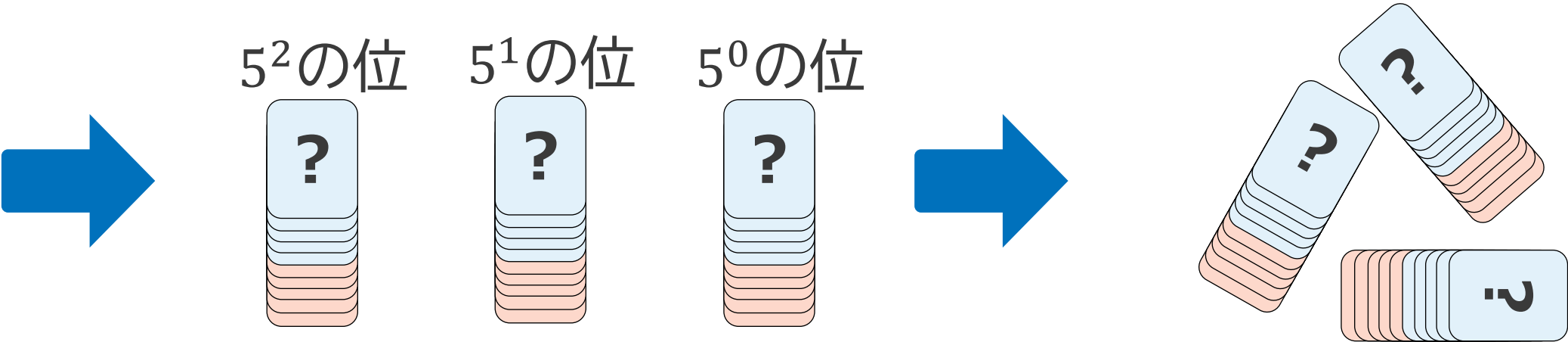
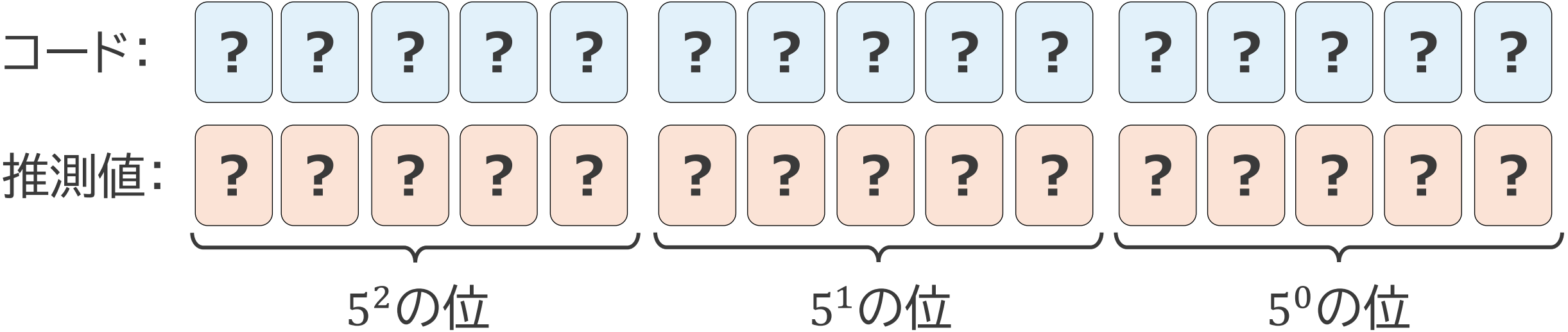
実行手順(3/5)

3. 各数字でカード束を作り、シャッフル(PSS)を行う



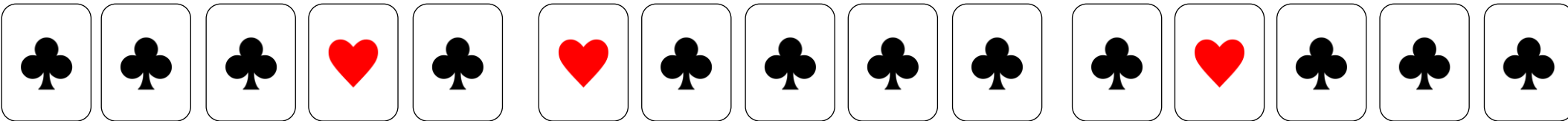
実行手順(4/5)

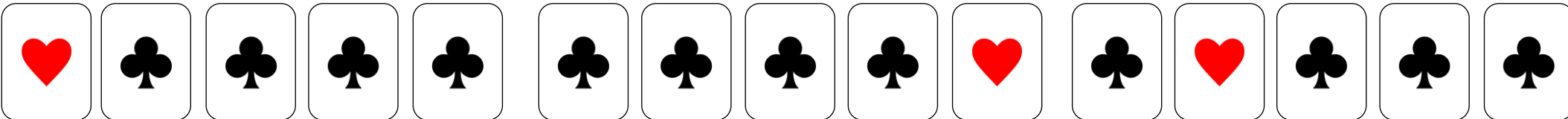
4. 各桁でカード束を作り、シャッフル(PSS)を行う



実行手順(5/5)

5. すべてのカードをめくり、判定を行う

コード: 

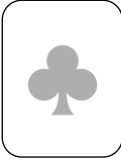
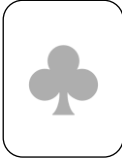
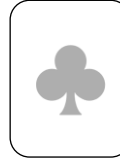
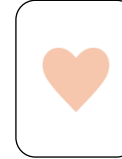
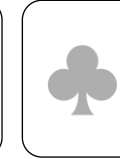
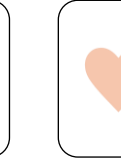
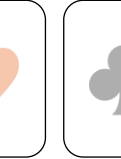
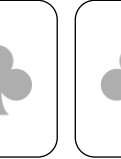
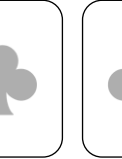
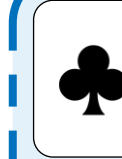
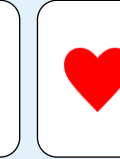
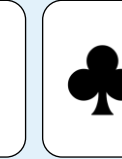
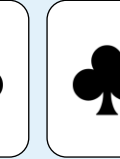
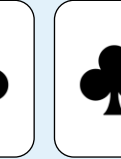
推測値: 

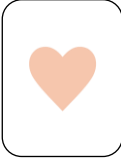
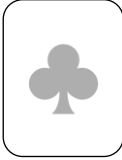
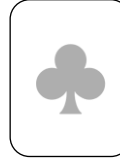
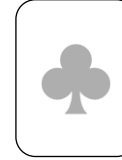
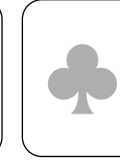
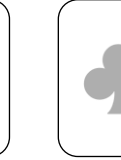
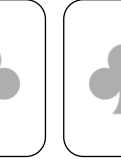
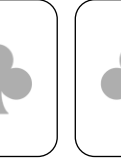
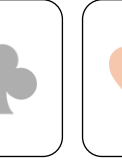
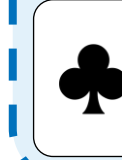
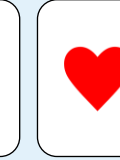
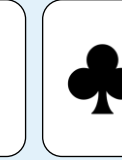
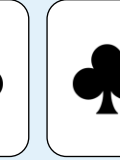
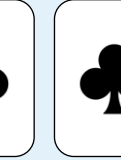
- コードと推測値でスートの位置が合っているかどうかを判定
 - 同じ桁でスートの位置が同じ場合は**Hit**
 - 異なる桁でスートの位置が同じ場合は**Blow**

コード: 304
推測値: 402

実行手順(5/5)

5. すべてのカードをめくり、判定を行う

コード:               

推測値:               

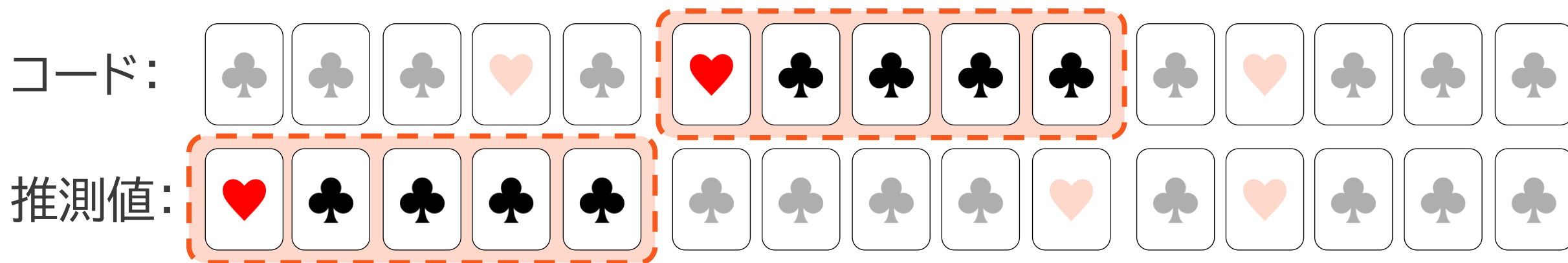
- コードと推測値でスートの位置が合っているかどうかを判定

- 同じ桁でスートの位置が同じ場合は**Hit**
- 異なる桁でスートの位置が同じ場合は**Blow**

コード: 304
推測値: 402

実行手順(5/5)

5. すべてのカードをめくり、判定を行う

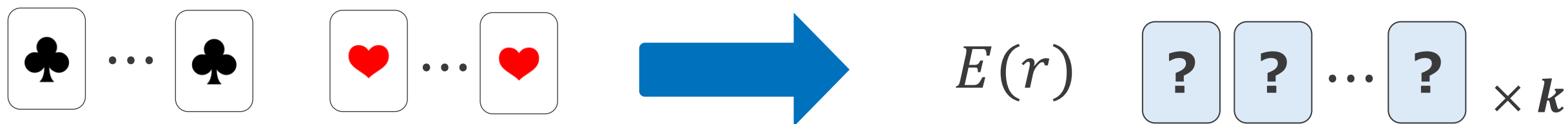


- コードと推測値でスートの位置が合っているかどうかを判定
 - 同じ桁でスートの位置が同じ場合はHit
 - 異なる桁でスートの位置が同じ場合はBlow

コード: 304
推測値: 402

コード生成プロトコルの概要

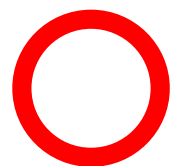
- 複数のコードをランダムに生成
- 生成されたコードの値は秘匿されている



※ r はランダムな値

- 同じ数字を複数の桁で使用してはいけない

例)



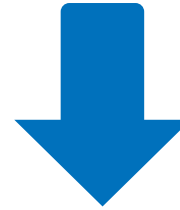
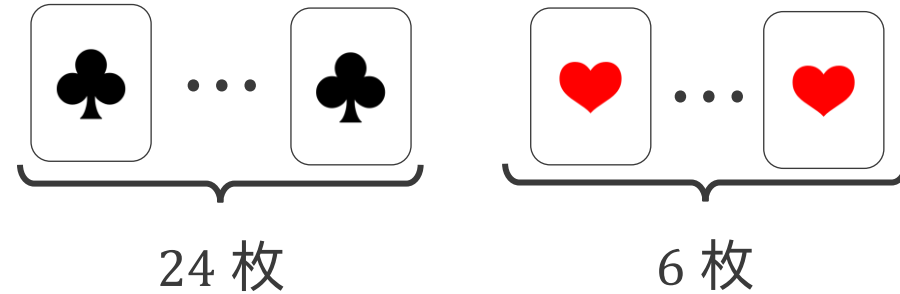
392



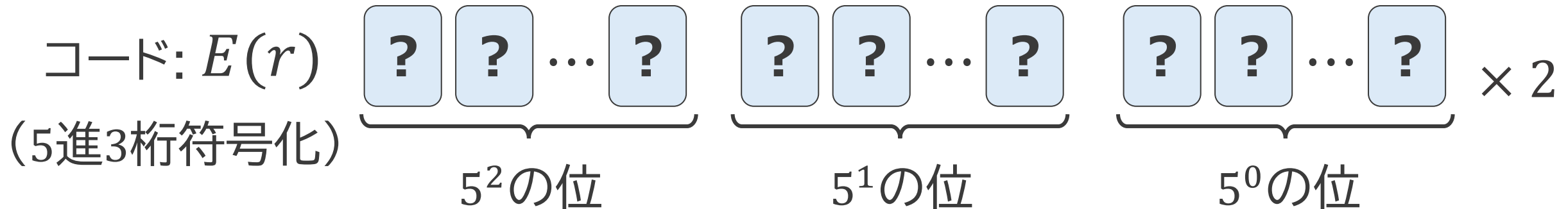
393

プロトコルの実行手順の例

コード2個の生成例

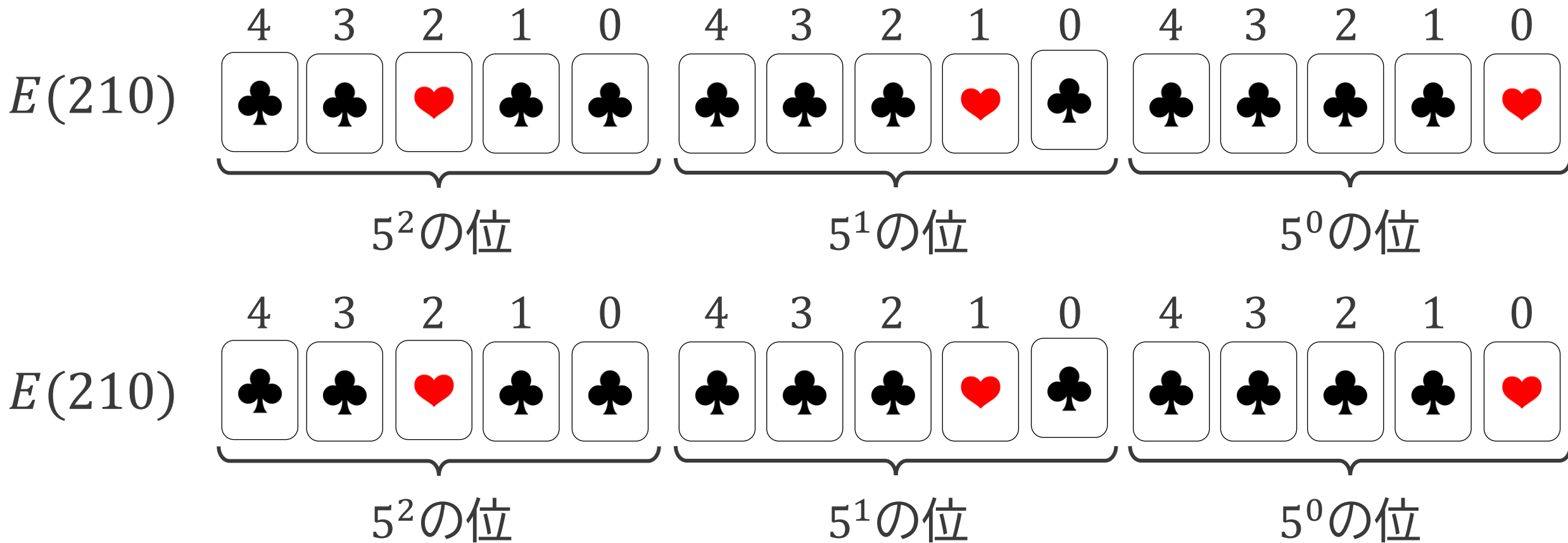


コード生成プロトコル



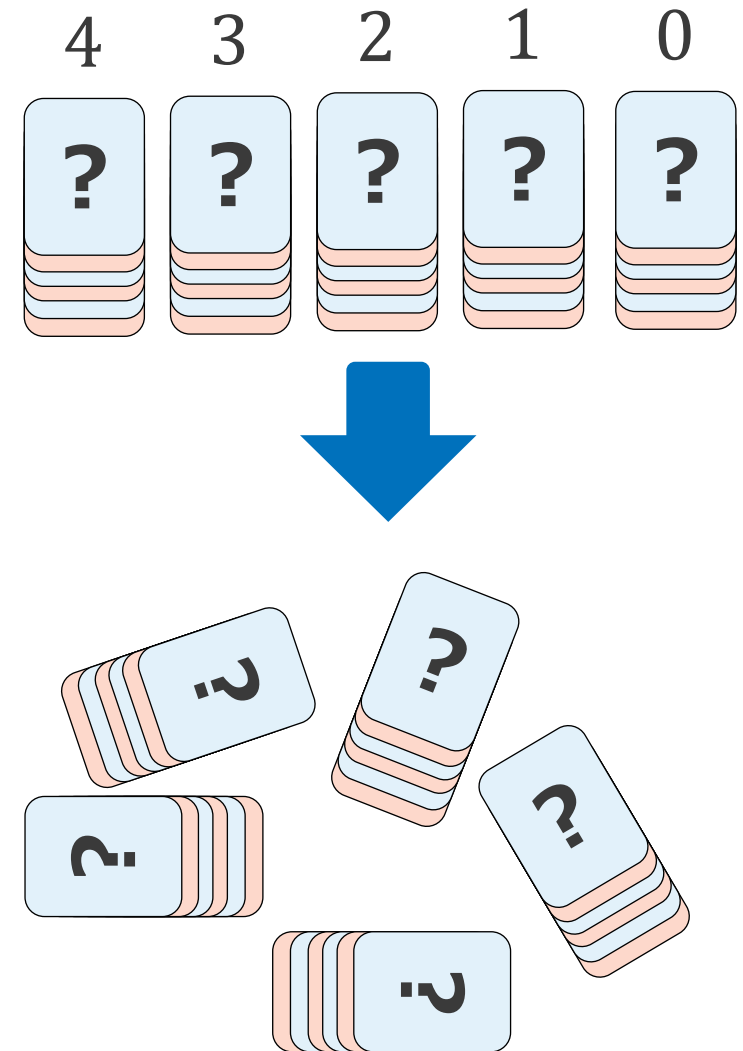
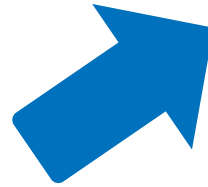
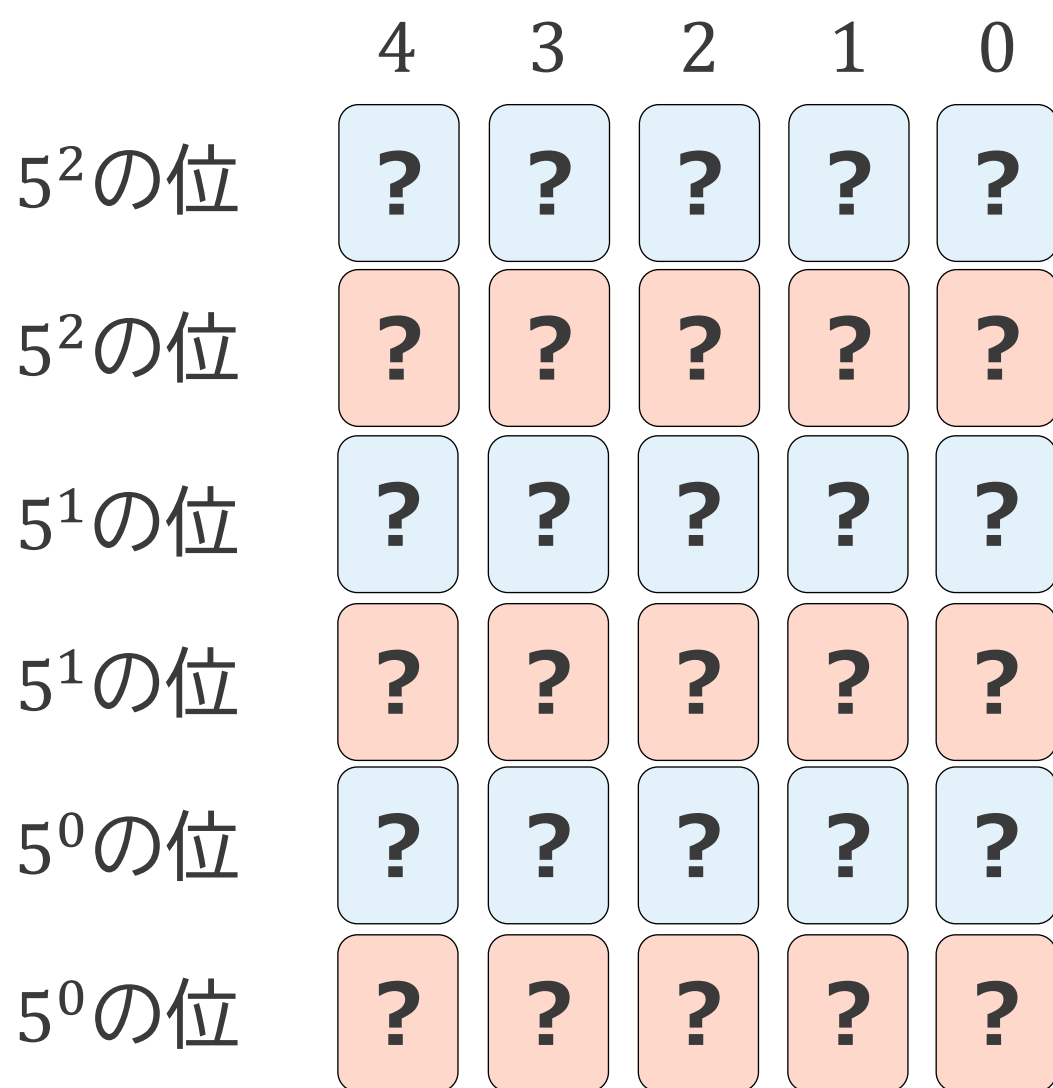
実行手順(1/3)

1. コード2個を入力として作る



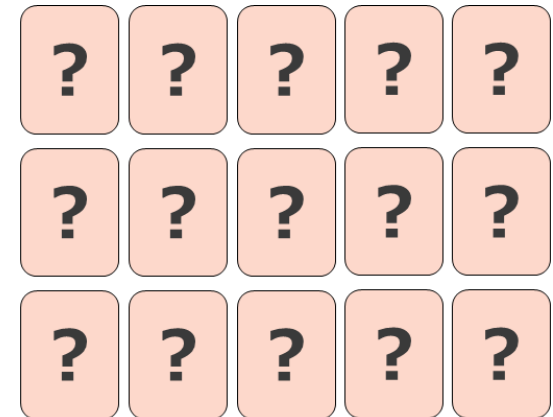
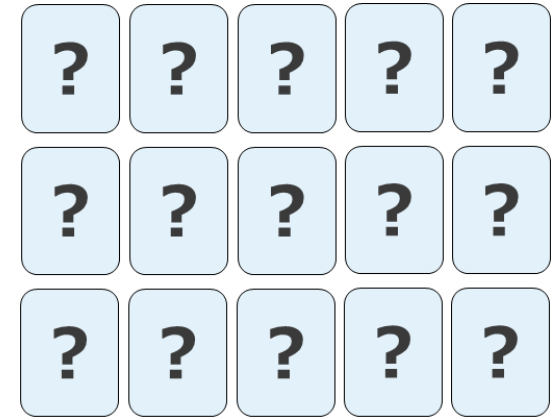
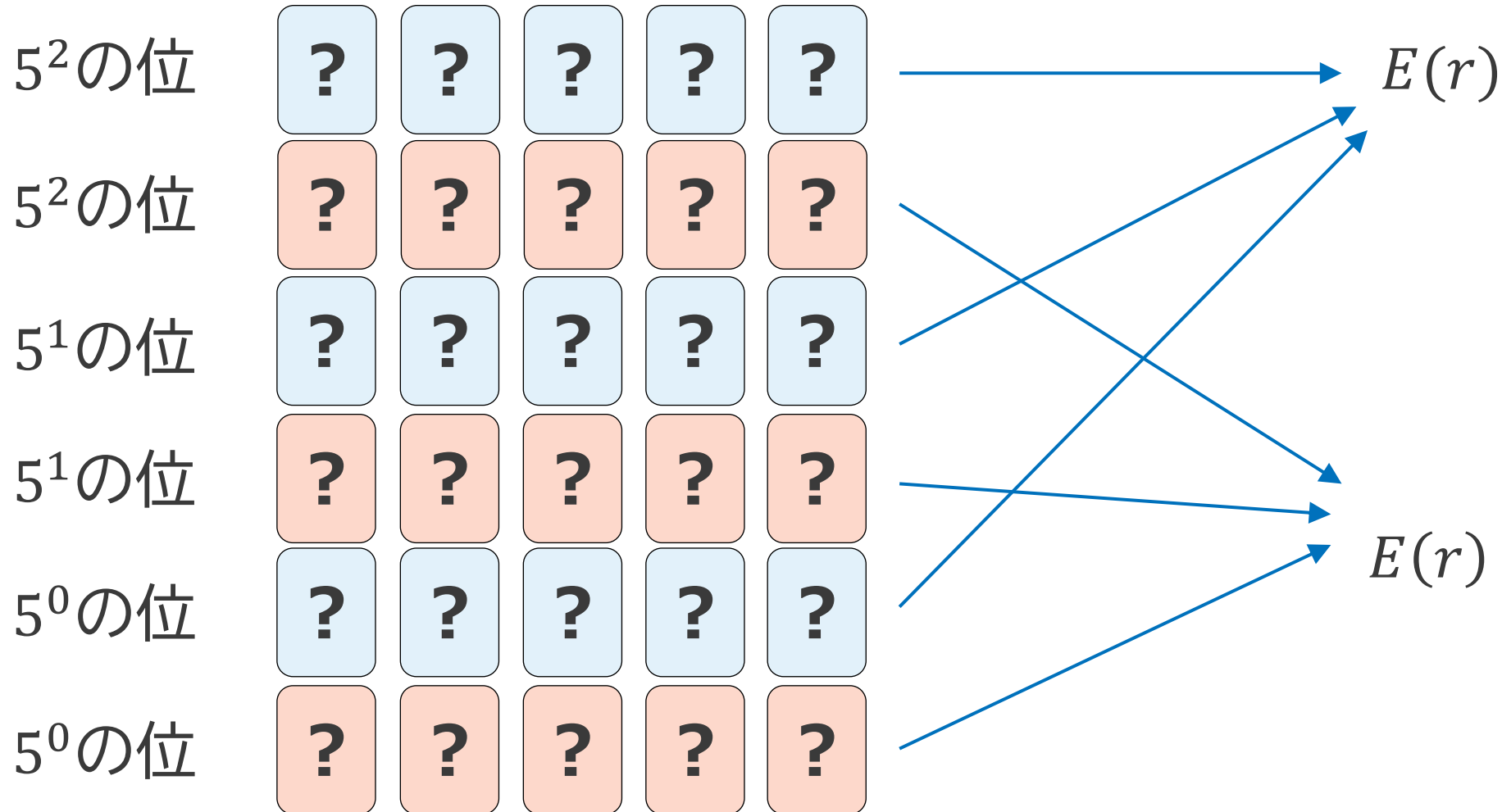
実行手順(2/3)

2. 各数字でカード束を作り、シャッフル(PSS)を行う



実行手順(3/3)

3. 各位から1個ずつカード束を取り、2つのコードを作成



一人で遊ぶ場合

通常の遊び方

- 1人(出題者不在)では遊ぶことができない
- 出題者がウソのヒントを返す場合がある

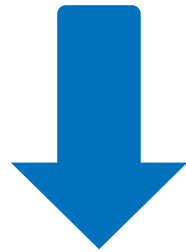
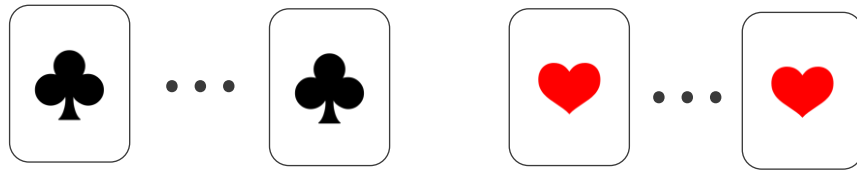
プロトコルを用いた遊び方

- 出題者が不在の場合でもプレイができる
- 解答者にウソのヒントが返されることがない
- コードは1回のシャッフルで生成できる

一人での遊び方(1/2)

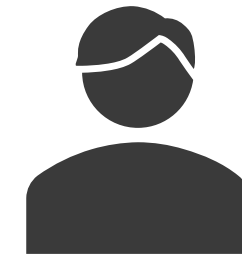
1. コード生成プロトコルを用いて k 個のコードをランダムに生成

※ k は最大ラウンド数を表す



コード生成プロトコル

コード: $E(r)$   ...  $\times k$



プレイヤー

一人での遊び方(2/2)

2. プレイヤーは推測値を設定し、フィードバックプロトコルを使用して **Hit** の数と **Blow** の数を入力

コード: $E(r)$? ? ... ? & 推測値



フィードバックプロトコル

Hit の数 & **Blow** の数



プレイヤー

3. ステップ2を繰り返し、プレイヤーが k ラウンド以内にコードを推測した場合、プレイヤーの勝利となる

まとめ

まとめ

- カードベース暗号のヒット & ブローへの応用
- 2種のプロトコルを提案
 - フィードバックプロトコル
 - コード生成プロトコル

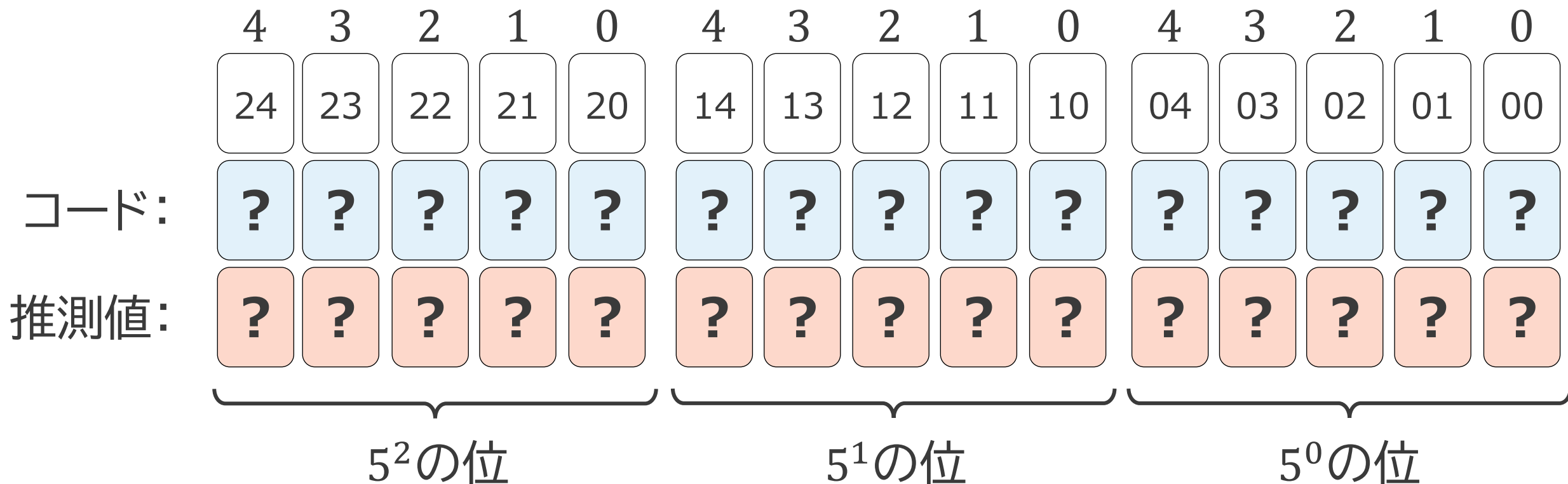
今後の研究課題

- プロトコルのデモを実践(研究室公開で実践)
- その他のゲーム(ポーカー・ブラックジャック)への応用

付録

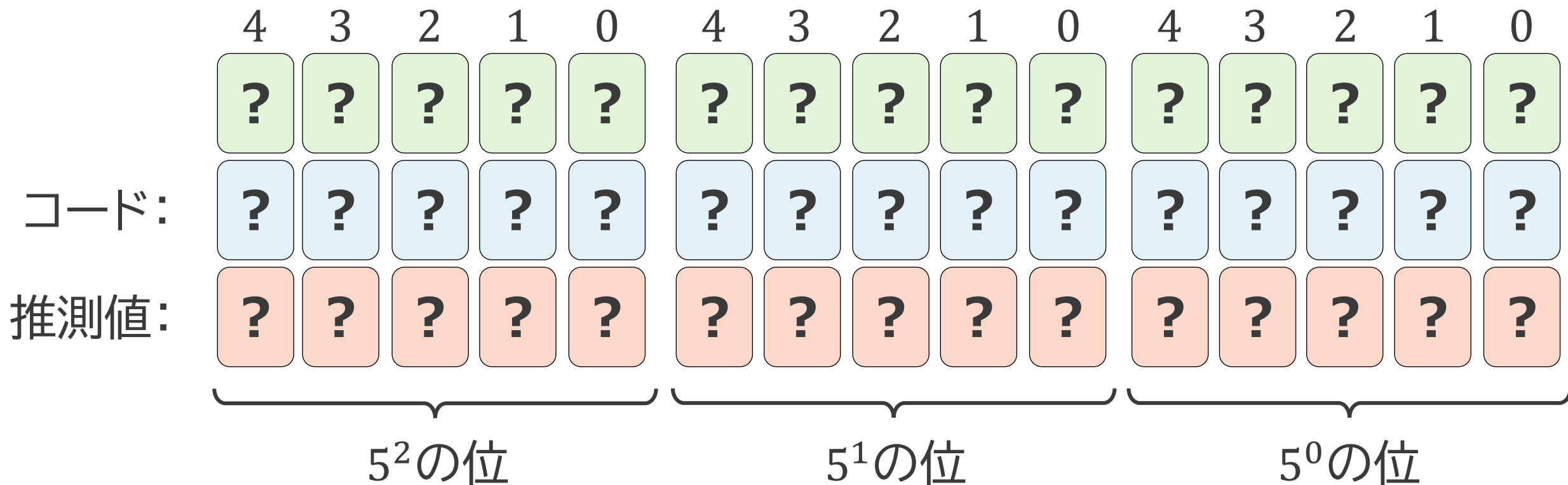
コード保存型プロトコルのアイデア

1. 数字カードを以下のように追加



コード保存型プロトコルのアイデア

1. 数字カードを以下のように追加



コード保存型プロトコルのアイデア

2. プロトコルを実行(追加した数字カードはめくらない)

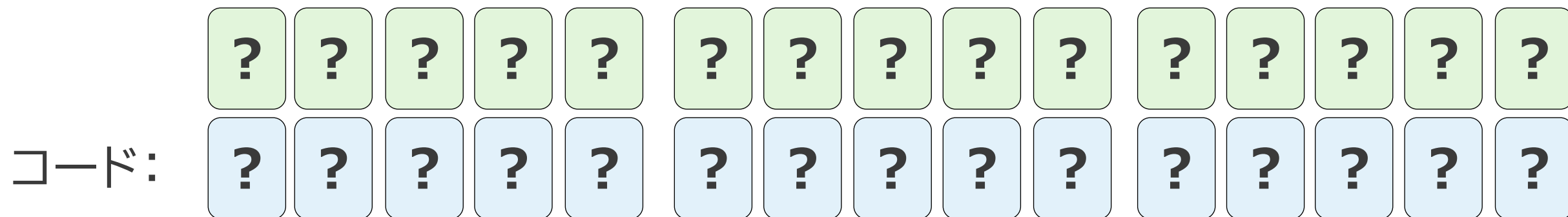
コード:

推測値:

?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
♣	♣	♣	♥	♣	♥	♣	♣	♣	♣	♣	♥	♣	♣	♣
♥	♣	♣	♣	♣	♣	♣	♣	♣	♥	♣	♥	♣	♣	♣

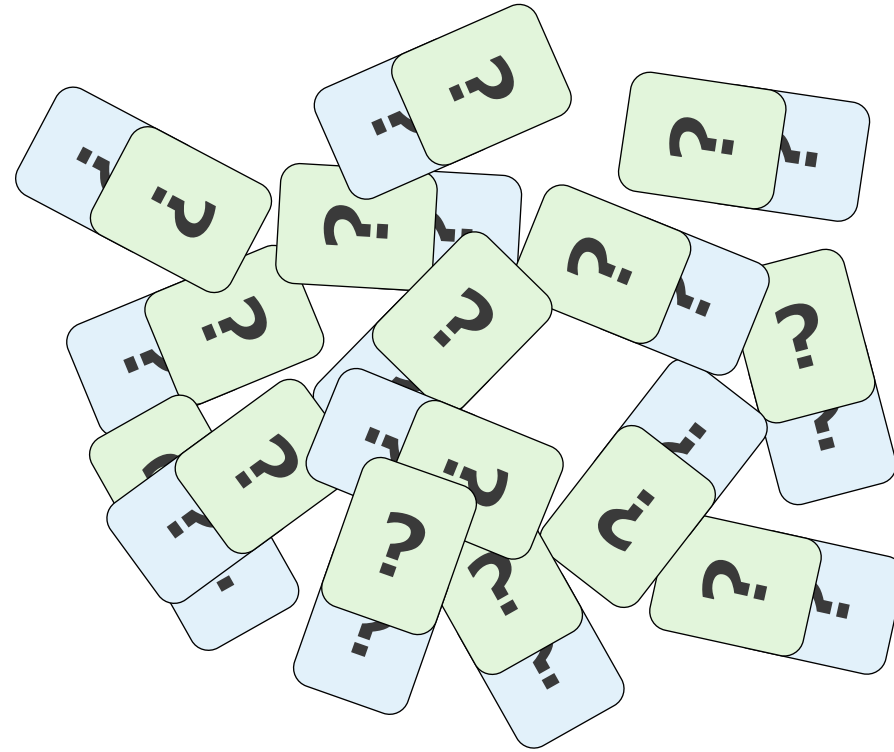
コード保存型プロトコルのアイデア

3. コードのカードをすべて裏にする



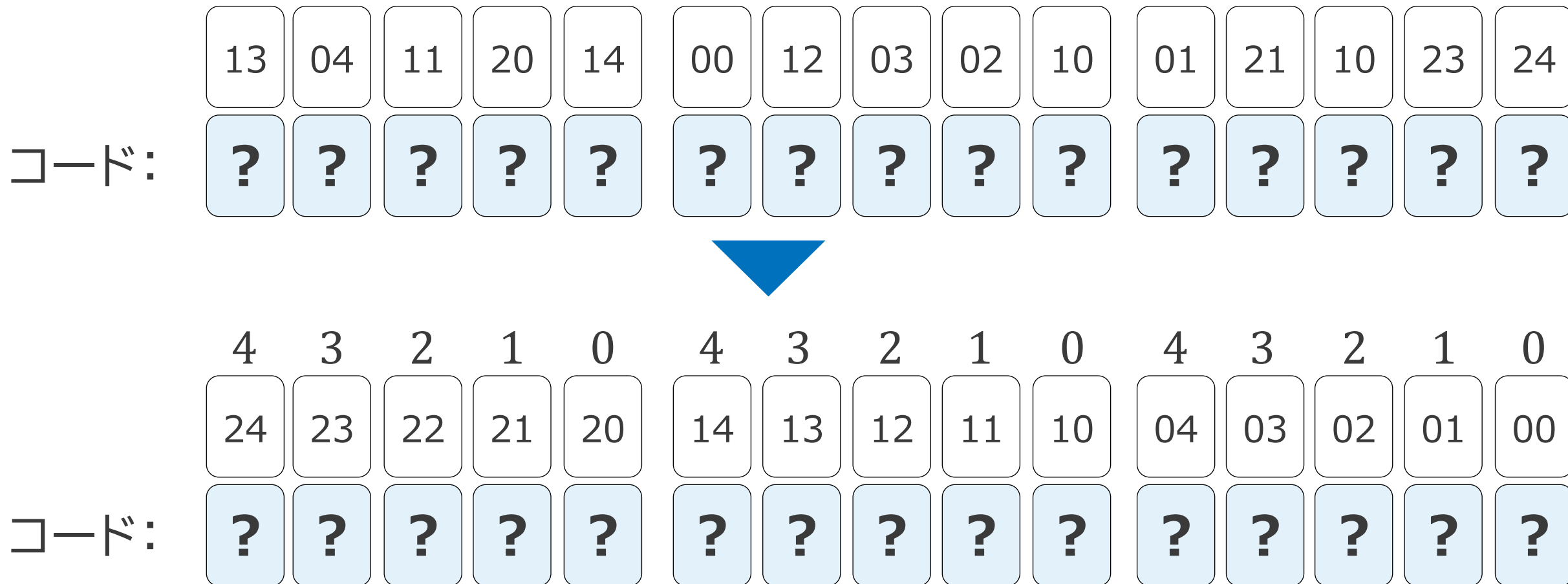
コード保存型プロトコルのアイデア

4. コードと数字カードを束にしてシャッフル



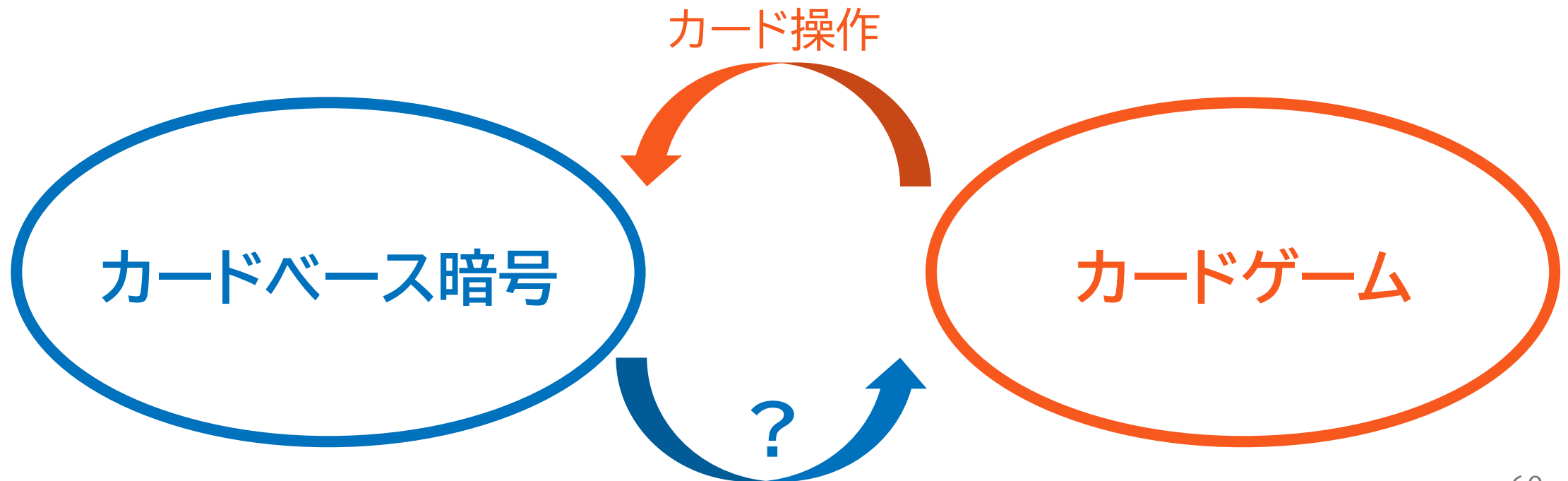
コード保存型プロトコルのアイデア

5. 数字カードをすべてめくり、数字に従って並び替える



カードベース暗号のゲームへの応用

- カードベース暗号はカードゲームをプレイする感覚に似ている
- カードベース暗号の技術をカードゲームに応用することは可能か？



ゲームへの応用の例

正規プレイヤーのシミュレーション



- プロトコルを用いてプレイを模倣
- ババ抜き[SMM24]UNO[SR25]など

ゲームマスターのシミュレーション



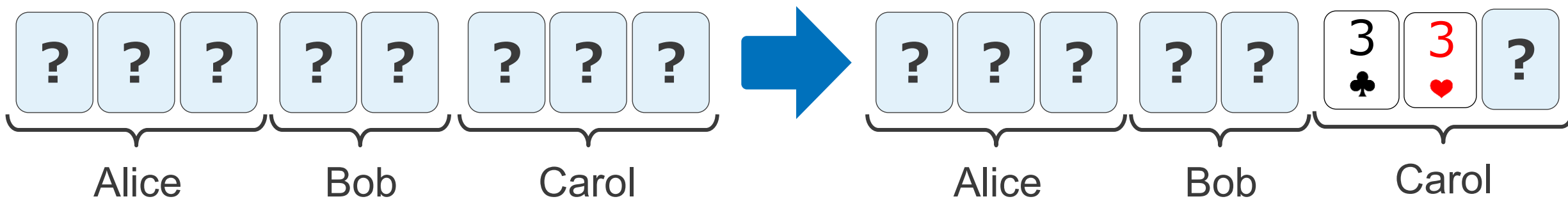
- 秘密情報の計算・判定を行う
- Gakmoro[MKH+25]など

正規プレイヤーの研究一覧

- ババ抜きでの正規プレイヤーの生成[SMM24]
- UNOでの正規プレイヤーの生成[SR25]
- 大富豪での正規プレイヤーの生成[MLMS26]
- Love Letterでの正規プレイヤーの生成[KM25a]

ババ抜きへの応用[SMM24]

- 2人でババ抜きを遊ぶ場合、ゲームが退屈になりがち
 - ジョーカーの位置が分かってしまうため
- キャロル(仮想プレイヤー)の手札管理プロトコルを提案



UNOへの応用[RS25]

- キャロル(仮想プレイヤー)の手札を秘密にしながら合法手を1枚ランダムに捨てるプロトコルを提案

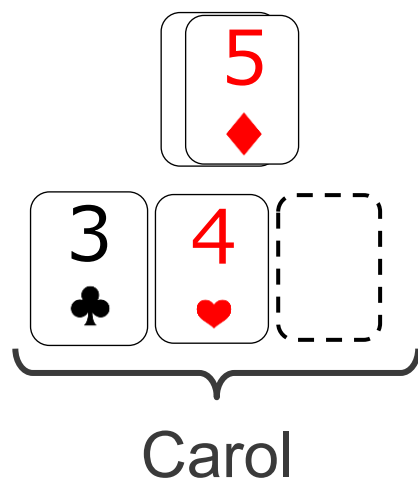
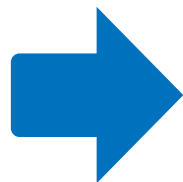
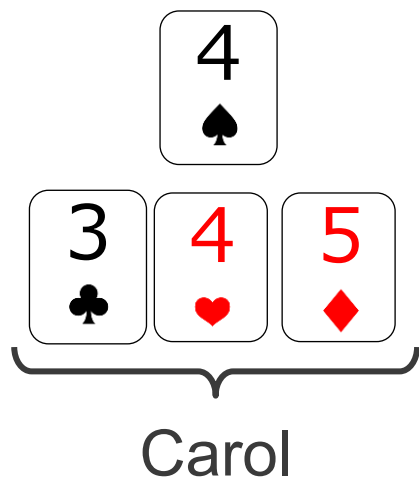
場札(捨て札の一番上)



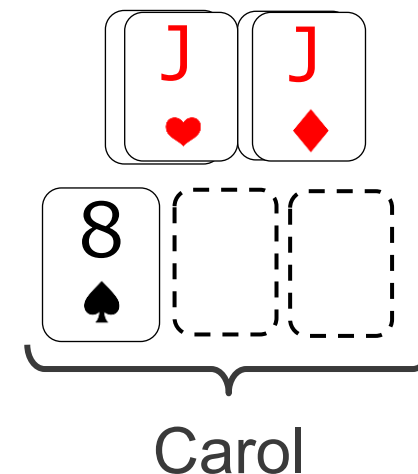
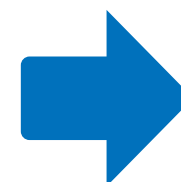
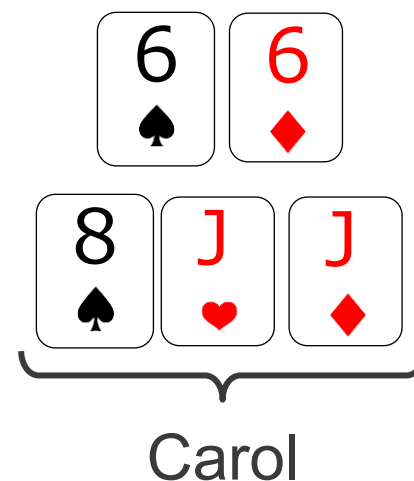
大富豪への応用[MLMS26]

- 手札を秘密にしながら、場札が2枚組、3枚組、4枚組の場合でも合法手をランダムに捨てられるようなプロトコルを提案

場札(捨て札の一番上)

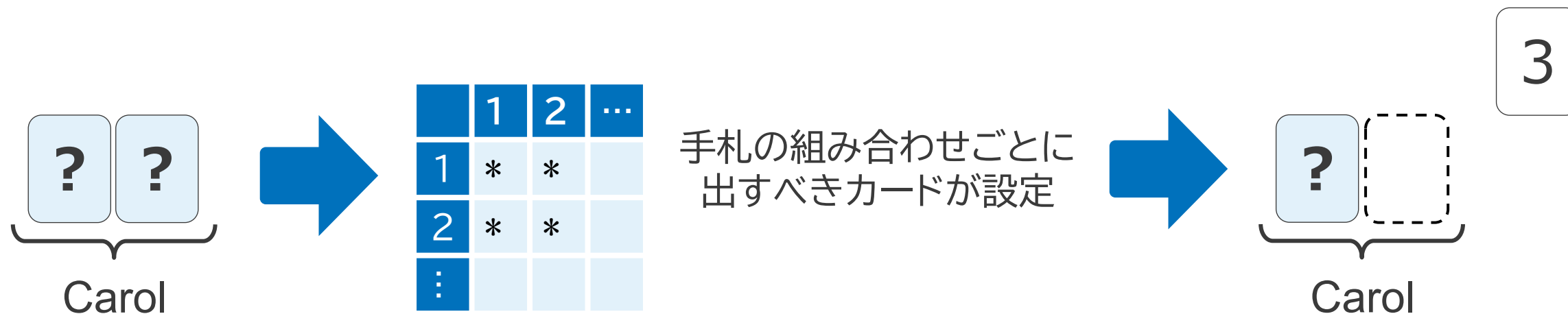


場札(捨て札の一番上)



Love Letterへの応用[KM25a]

- Love Letterはプレイヤーが1～8までの数字カード1枚の手札を持ち、手番で山札からカードを1枚引き、手札2枚から1枚を捨てる
 - 捨てた数字カードによって効果が異なる
- 仮想プレイヤーが手札を秘密にしながら、戦略表に従ってカードを捨てるプロトコルを提案

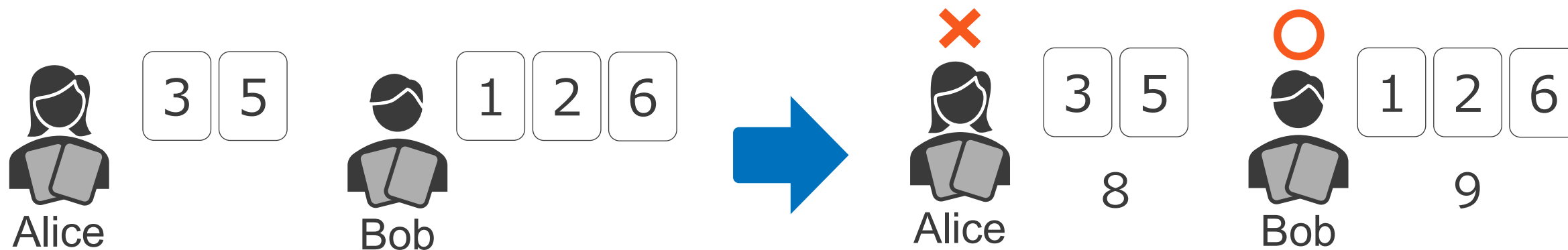


ゲームマスターの研究一覧

- GaKmorroでのゲームマスター生成[MKH+25]
- Werewolf(人狼)でのゲームマスターの生成[KKO26]
- Tagironでのゲームマスター生成[KM25b]

Gakmoroへの応用[MKH+25]

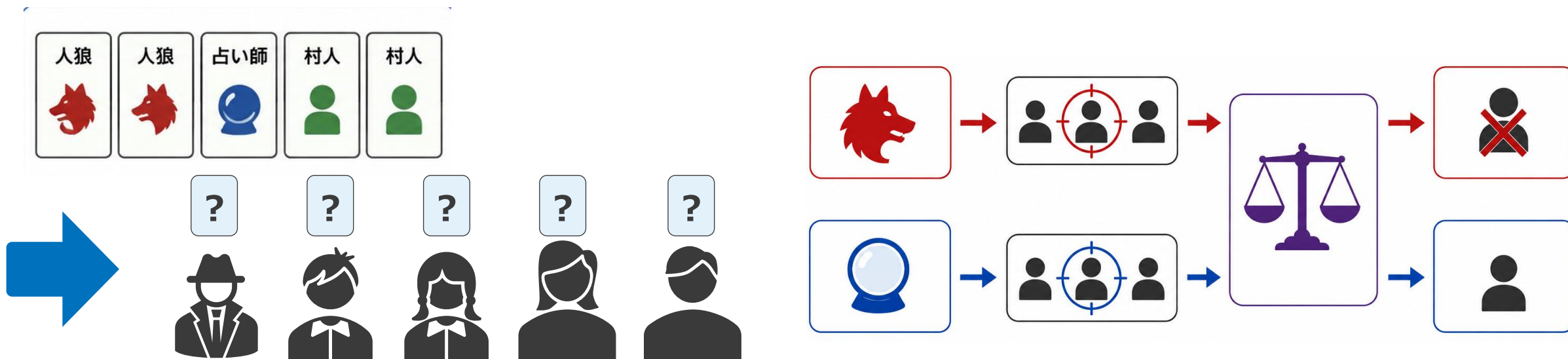
- 2人のプレイヤーが1から7までの数字が書かれた7枚のカードから毎ターン秘密裏に1～3枚を提出し、合計値を競うゲーム
- プロトコルは以下の操作を行う
 1. 各プレイヤーが1～3枚のカードを提出しているか確認
 2. 提出されたカードの値の合計値を計算
 3. 合計値や提出枚数を秘密にしたまま勝者のみを発表



人狼への応用[KKO26]

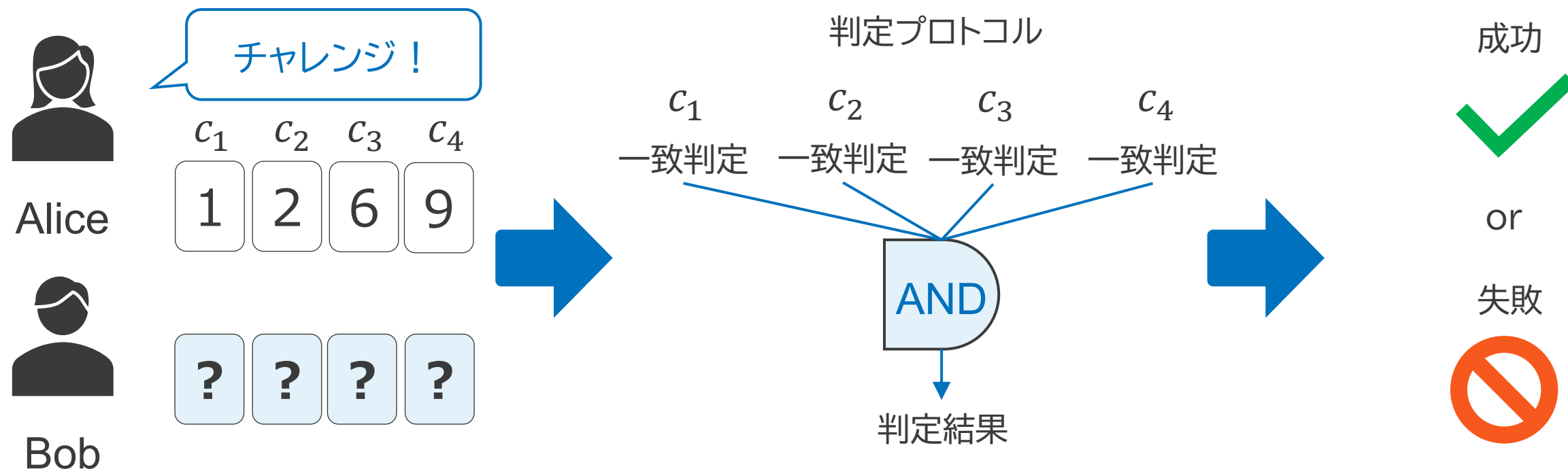
プロトコルは以下の操作を行う

1. 役職の秘密裏の割り当て
2. 夜のアクションの調停(人狼による襲撃や占い師による確認)
3. 情報のアナウンス



Tagironへの応用[KM25b]

- プレイヤーが他プレイヤーの4枚の数字カードを推測するゲーム
- 推測(チャレンジ)が合っているかどうかだけを判定するプロトコルを提案



その他の研究

- 手札の不正検知をするプロトコル[KO22a]
- 金持ち比べプロトコルを用いたゲーム[KO22b]
- 秘匿数拳ビンゴプロトコル[iKM25]

[KO22a] K.Koizumi and M.Ohtsuji, Proposition of New Secure Multiparty Computation Protocols for Unfair Player Using a Deck of Cards, SCIS 2022.

[KO22b] K.Koizumi and M.Ohtsuji, 対戦カードゲームにおいて強さを公開することなく勝敗のみを知りゲームをよい面白くできる可能性のある新しい秘密計算カードプロトコルの提案, IPSJ-GI 2022.

[iKM25] R.Igari, Y.Komano and T.Mizuki, Suken BINGO: An Application of Card-Based Cryptography to Psychological Board Games, SecITC 2025.