

問題解決技法入門

2. 社会的選択：安定性 安定結婚問題

堀田 敬介

浮気しない？カップル

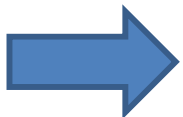
- 6人の男女がいます。少子化対策？のため、6組のカップルを作り結婚させちゃいましょう。でも各自の**好き嫌い**を考えずに強引にくっつけちゃうと、**浮気する人**が出るかもしれません。浮気しないように6組のカップルをつくれますか？



どうすれば浮気しないの？

浮気しないってどういうこと？

浮気ってどういう状況で起こる？

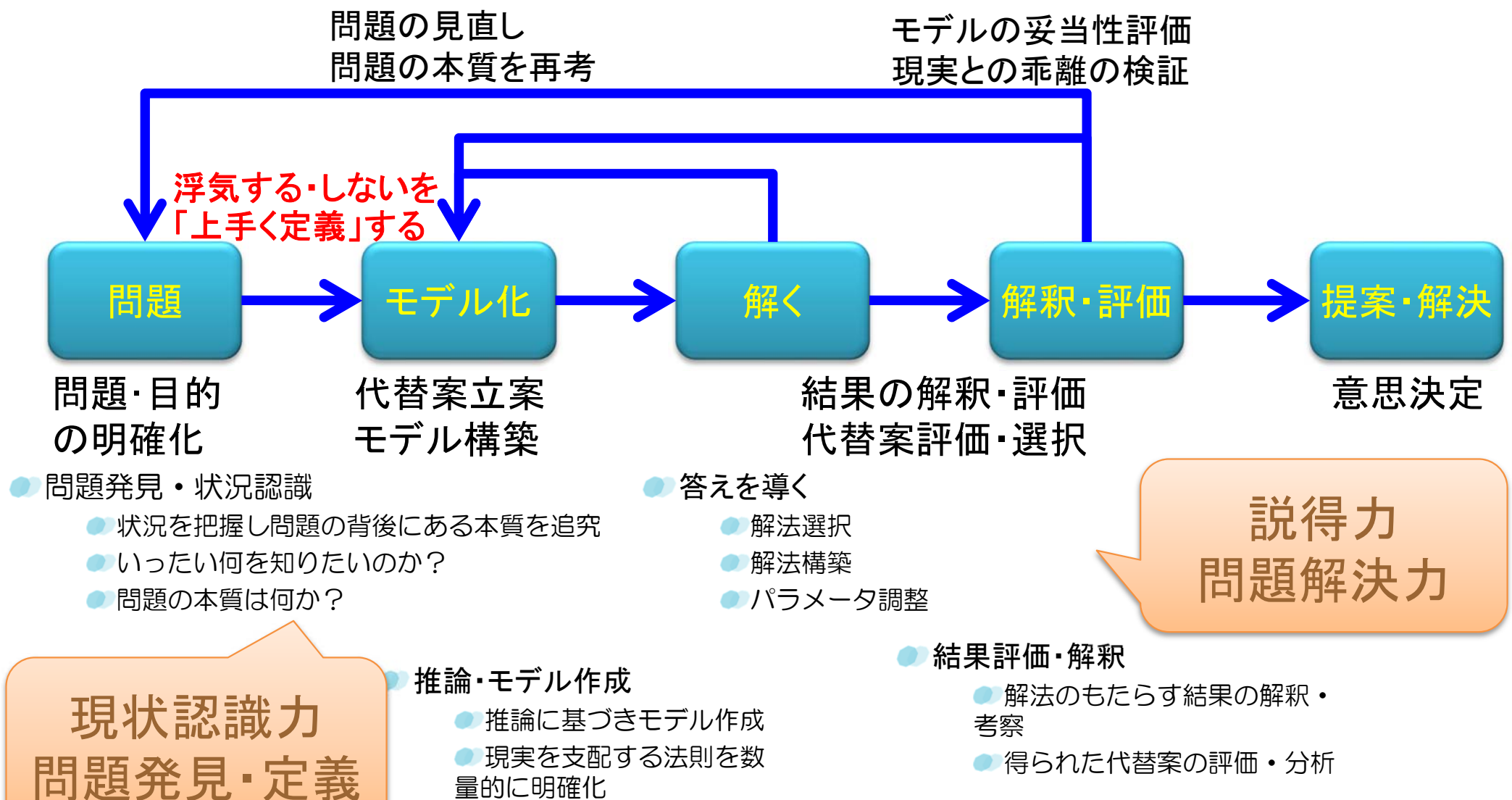


浮気する・しないを「**上手く定義**」する

問題解決

論理的思考力
データ分析, 統計学
数理的アプローチ

「問題の把握」から「意思決定」までの流れ



安定結婚問題

- n 人の男性の集合と, m 人の女性の集合が存在し, 各人は異性全員の選好順序をもっている. このとき, 安定なマッチングを見つけない.

安定マッチング

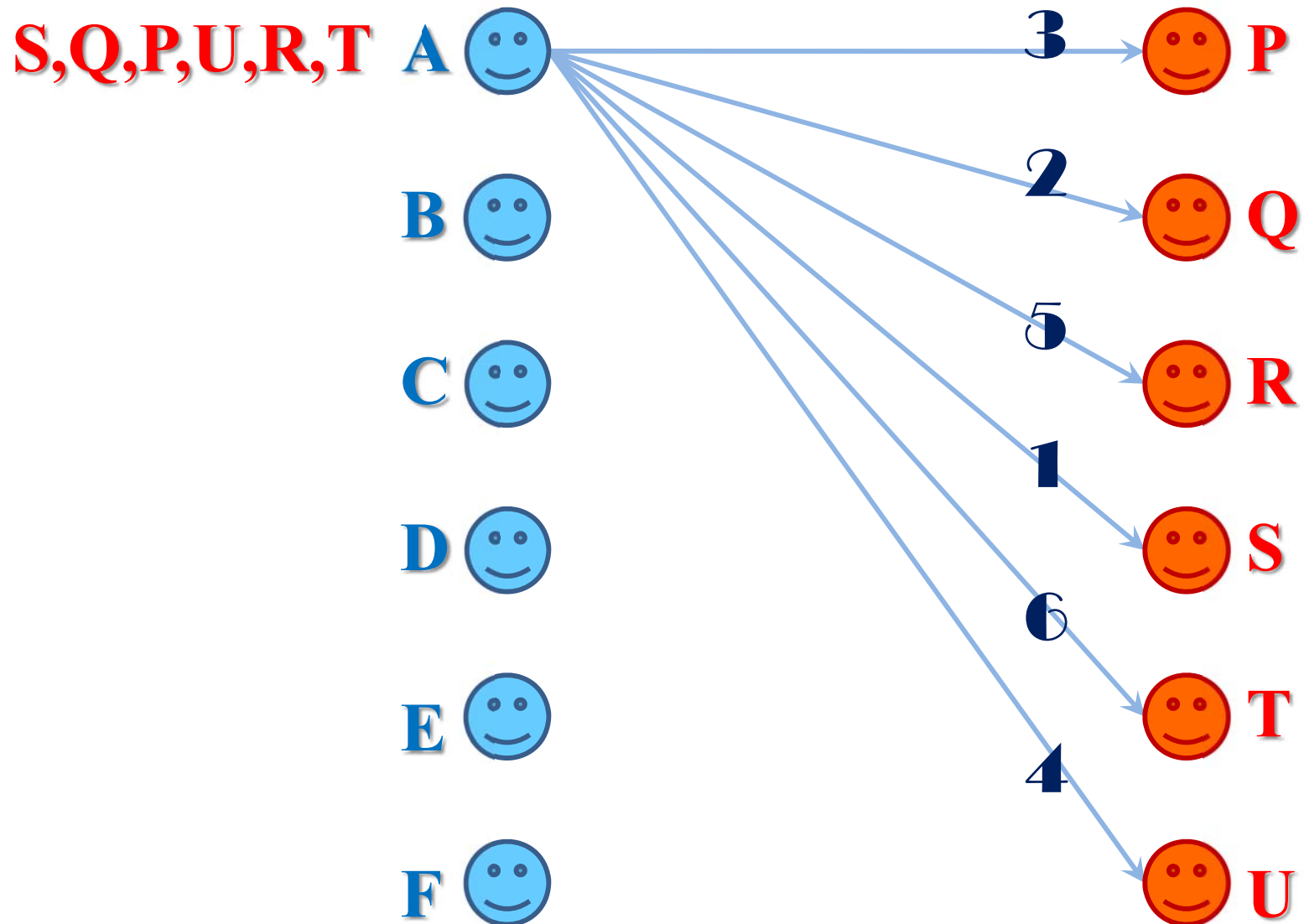
浮気できない

不安定なマッチング

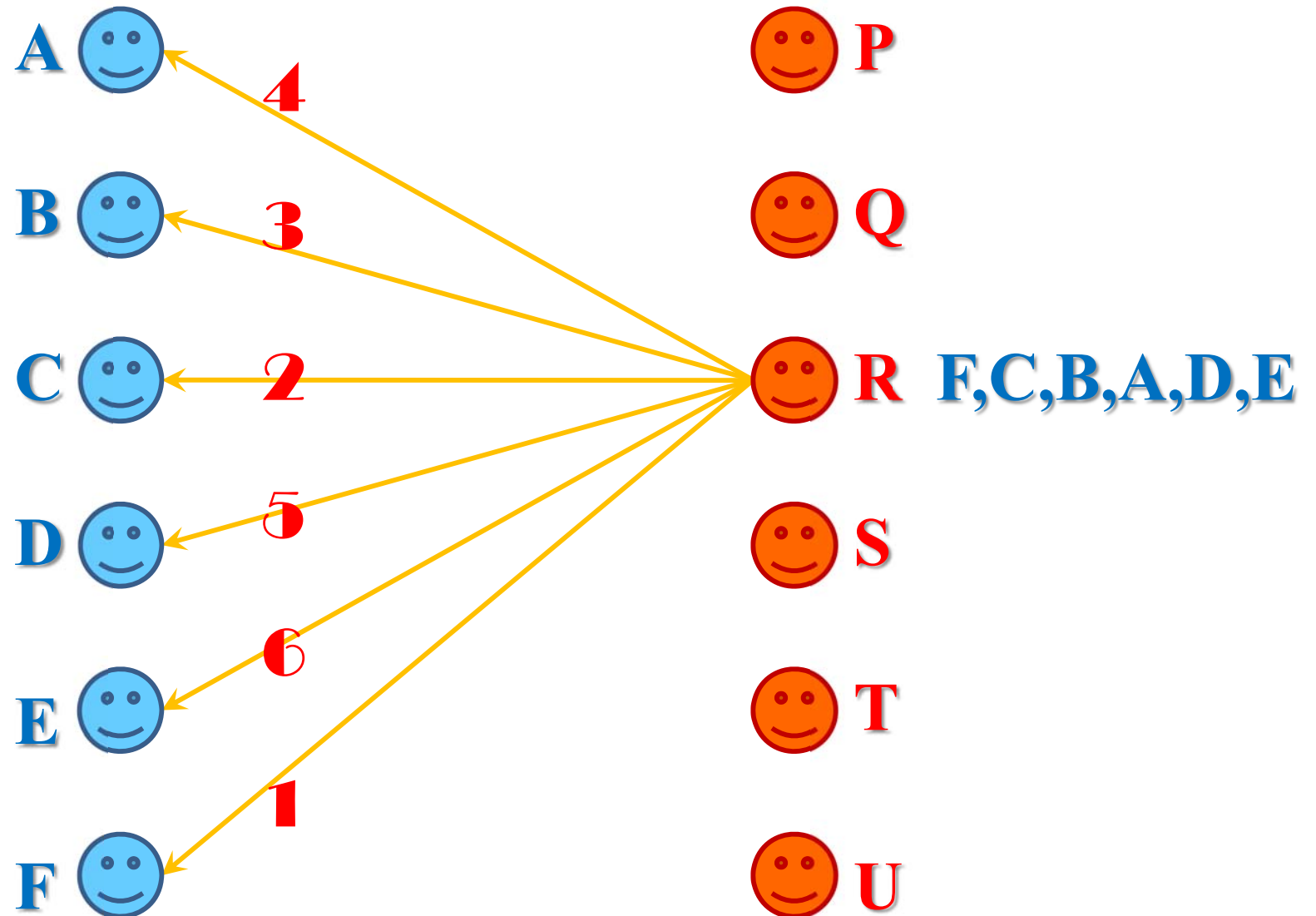
浮気できる



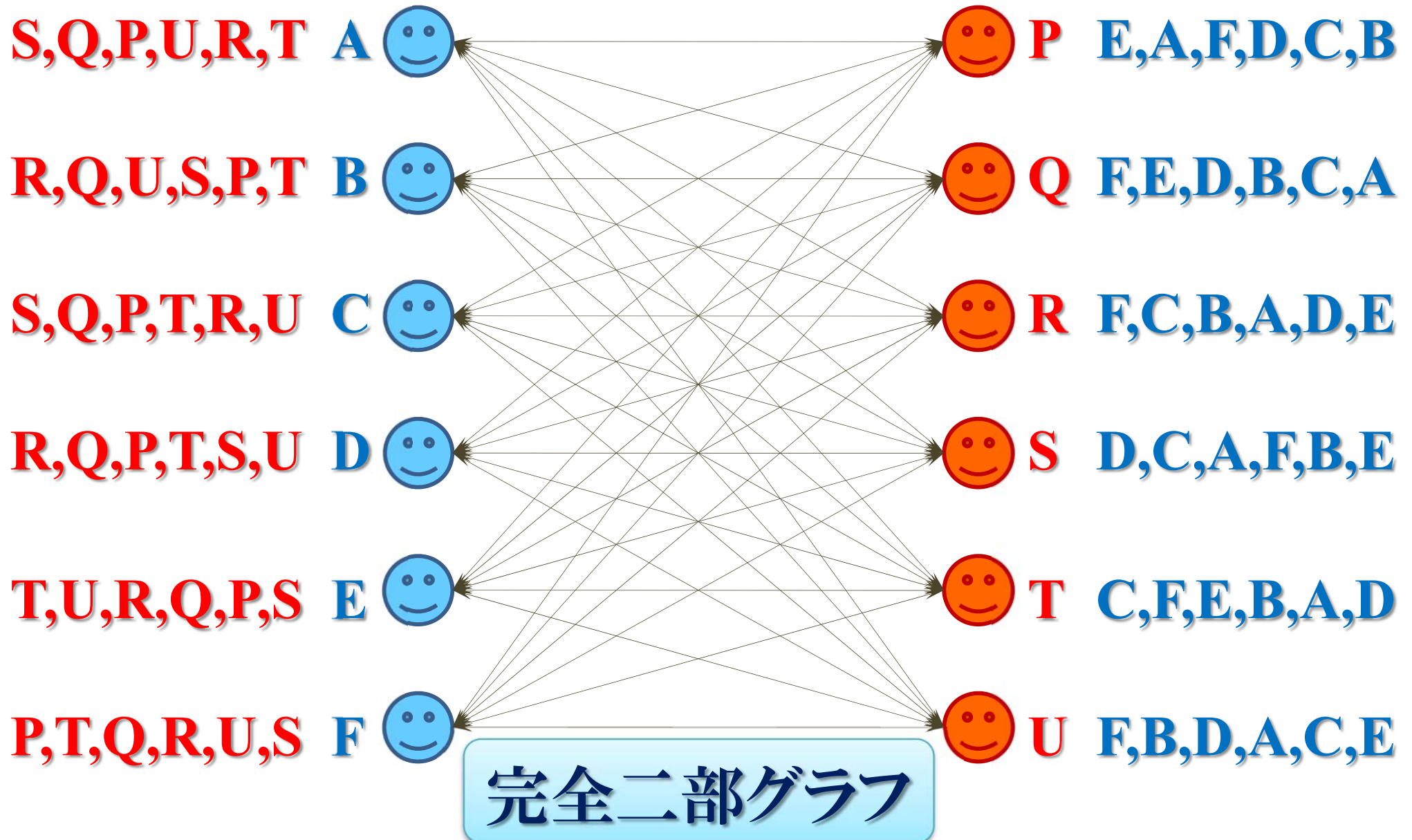
安定結婚問題（各自の選好順序）



安定結婚問題（各自の選好順序）

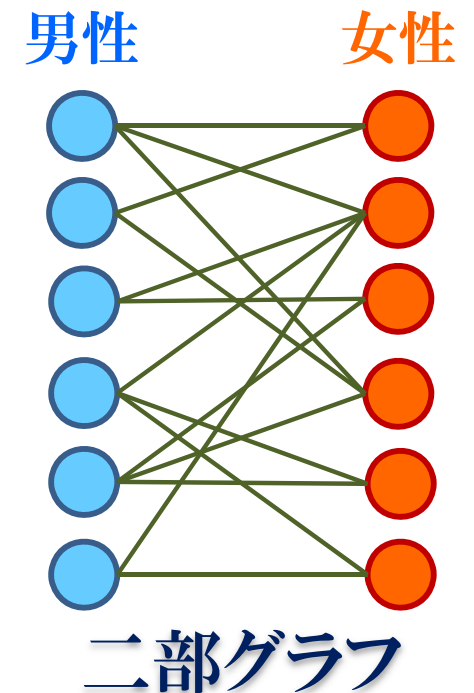
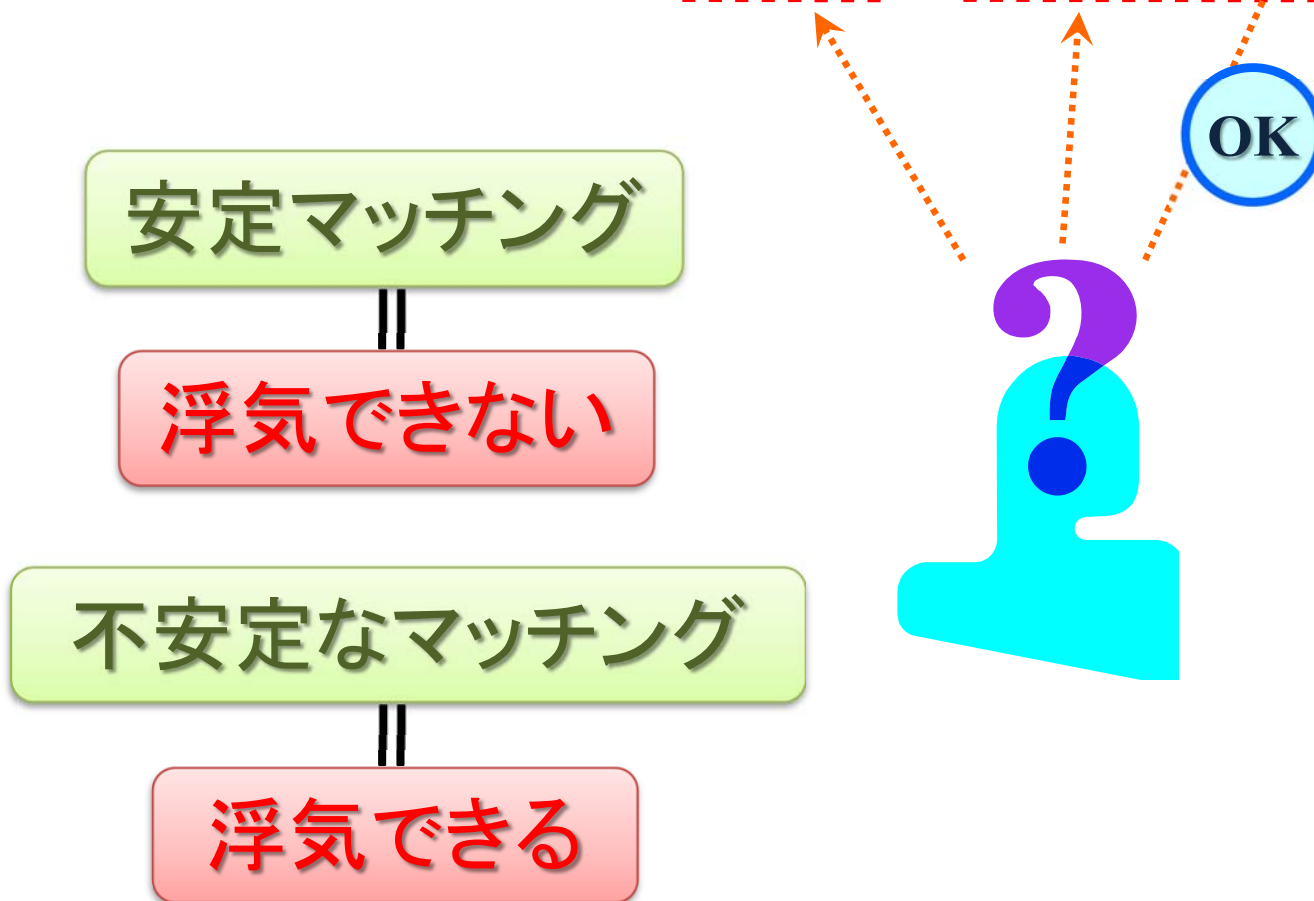


安定結婚問題（各自の選好順序）



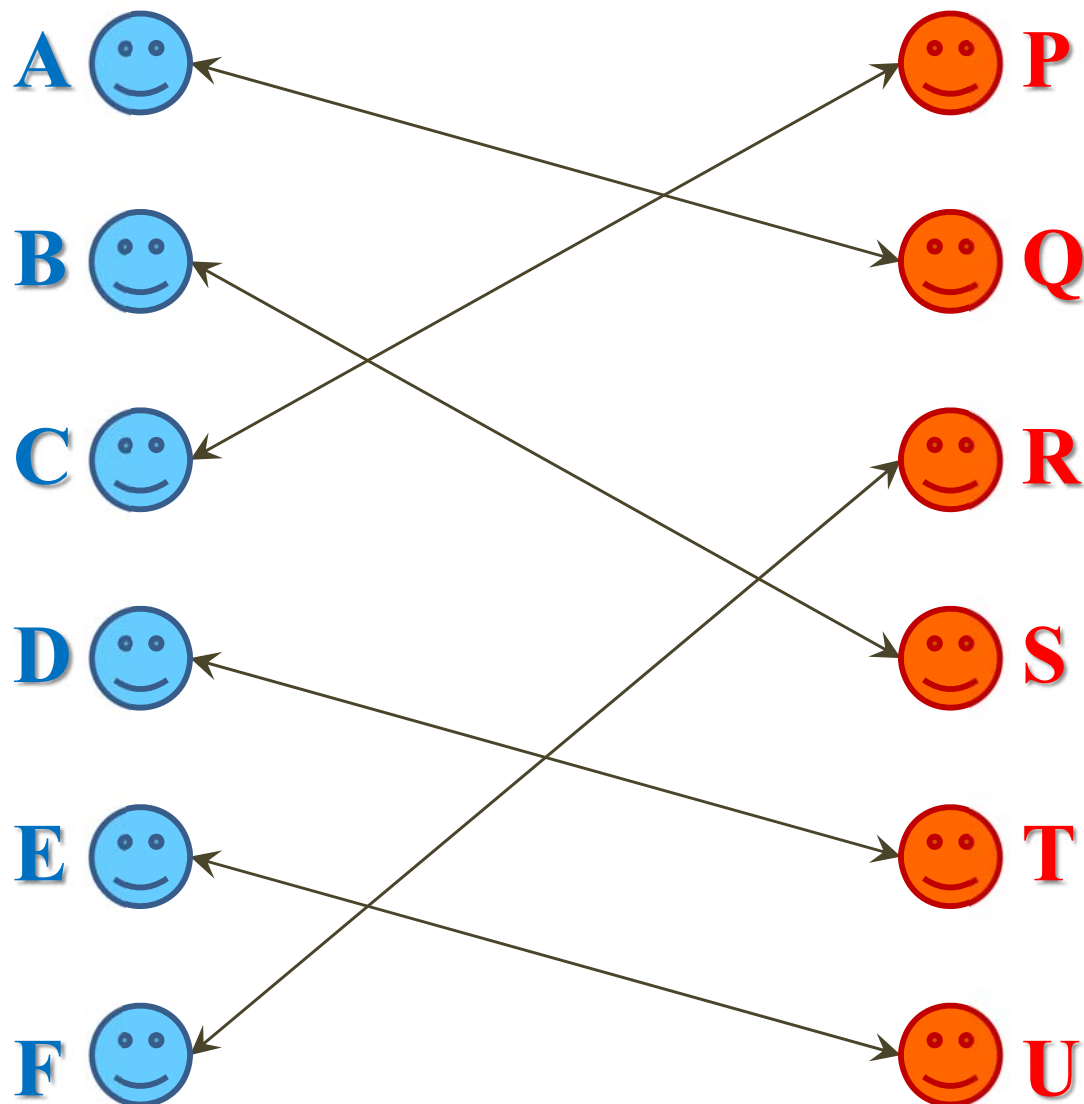
安定結婚問題

- n 人の男性の集合と、 m 人の女性の集合が存在し、各人は異性全員の選好順序をもっている。このとき、安定なマッチングを見つけない。



男性：点集合 V_1
女性：点集合 V_2

安定結婚問題(マッチング)



マッチング

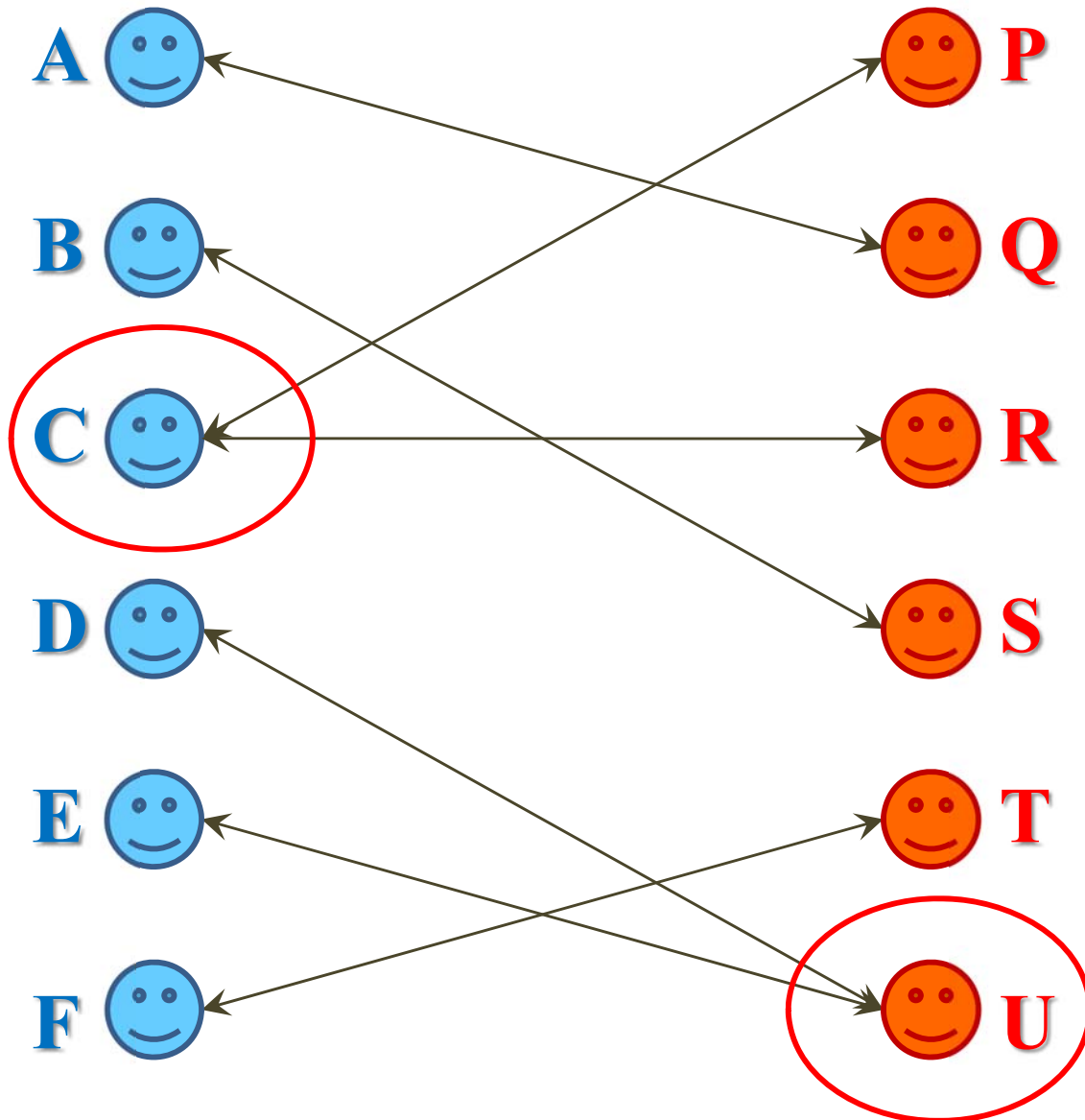
端点を共有しない枝の集合

つまり, どの点 (node) も
高々1本の枝 (edge) にのみ
接続 (incident to) している

完全マッチング

全ての点 (node) が, マッチ
ング (matching) の枝 (edge)
に接続しているとき, その
マッチングを完全マッチング
という

安定結婚問題 (マッチング)



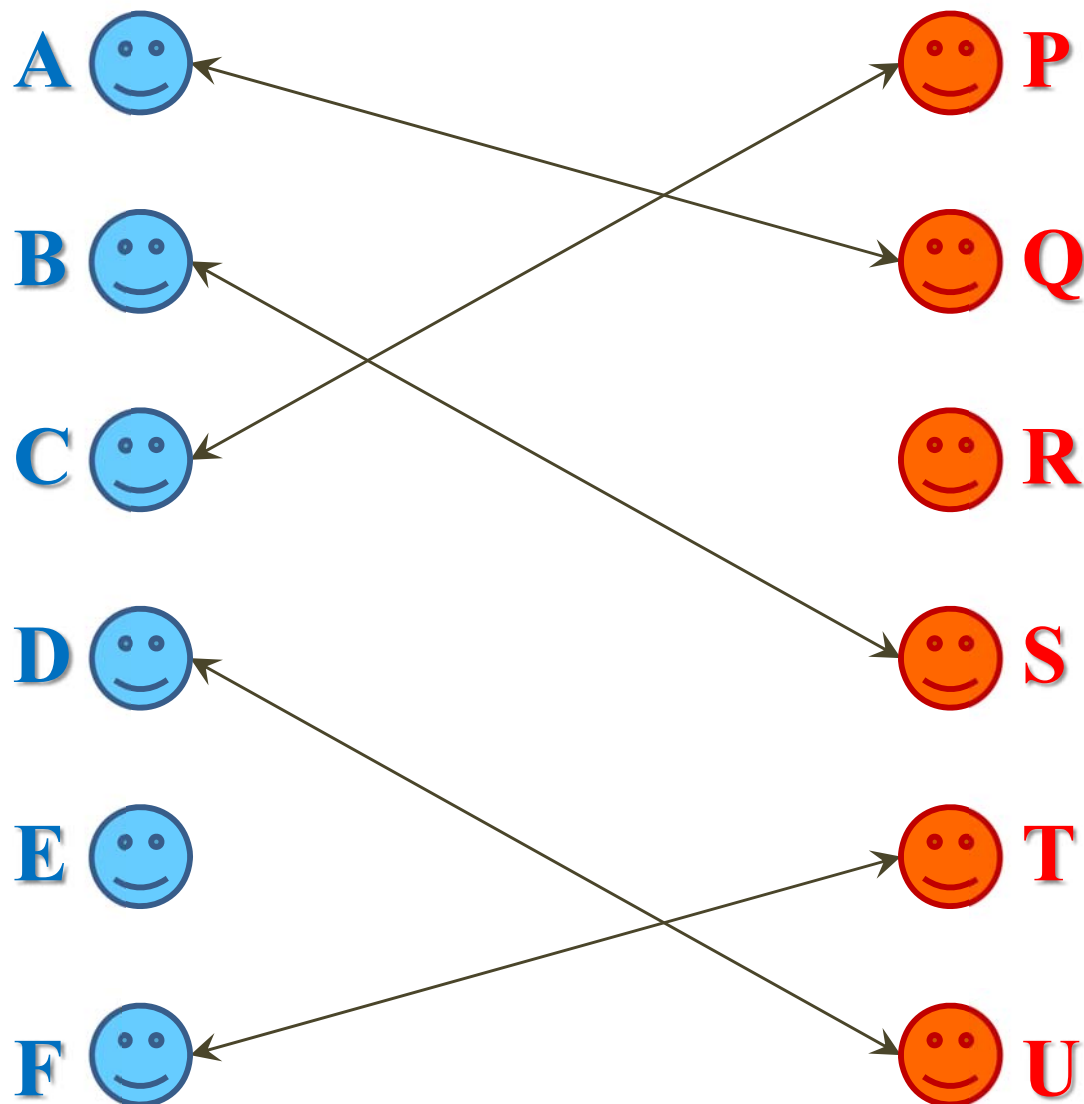
この枝集合は、マッチング
(matching)ではない

なぜだかわかる？

その通り！ マッチングで
はありません。

なぜなら、端点を共有する
枝がある(二股をかけてい
る人がいる)から

安定結婚問題(マッチング)



この枝集合は、マッチング
(matching)だろうか？

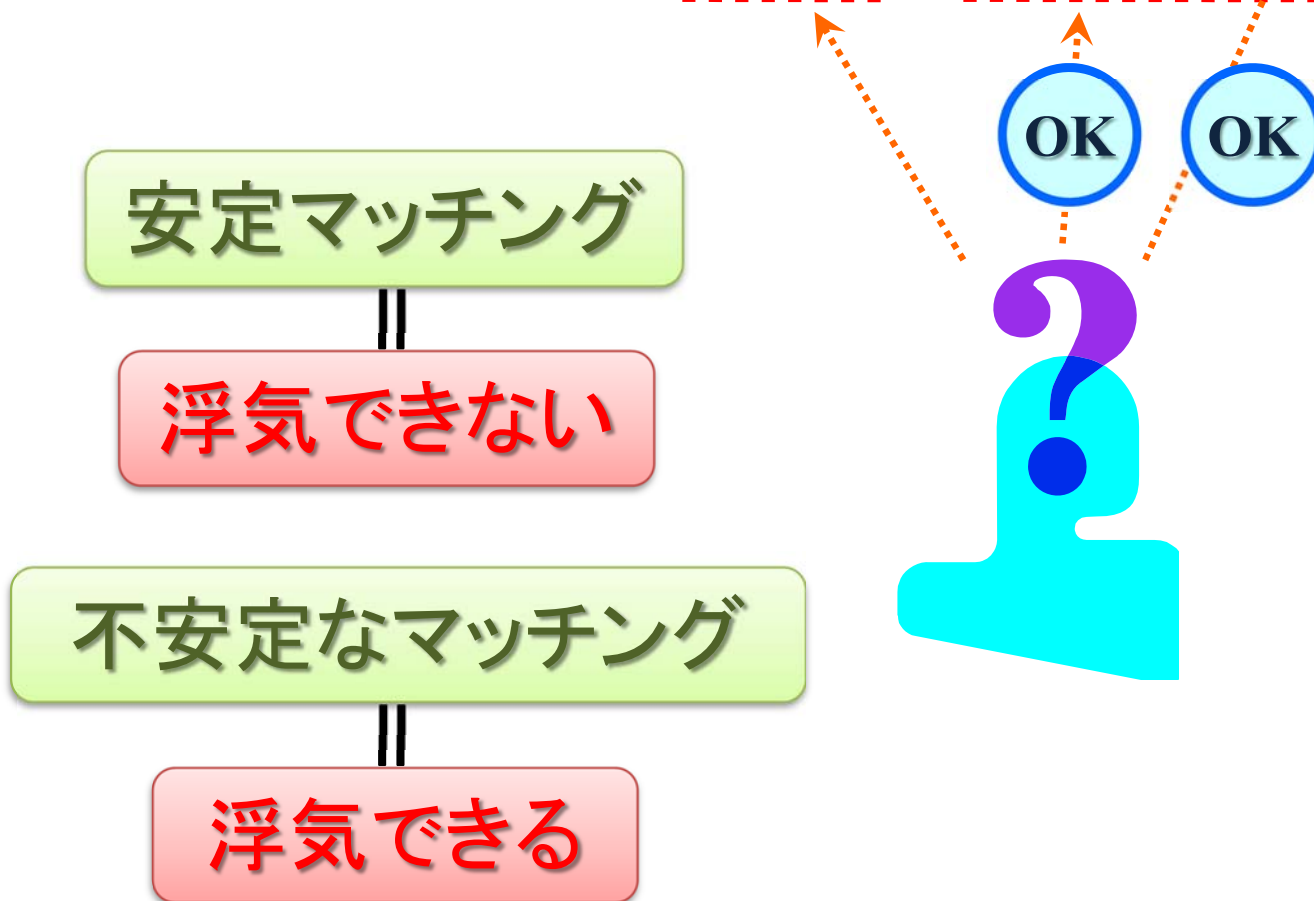
マッチング (matching) です。
でも、完全マッチング
(perfect matching) ではない
ので、ペアを組んでない人
がいるね。

つまり、我々は完全マッチング
を求めたいのだよ

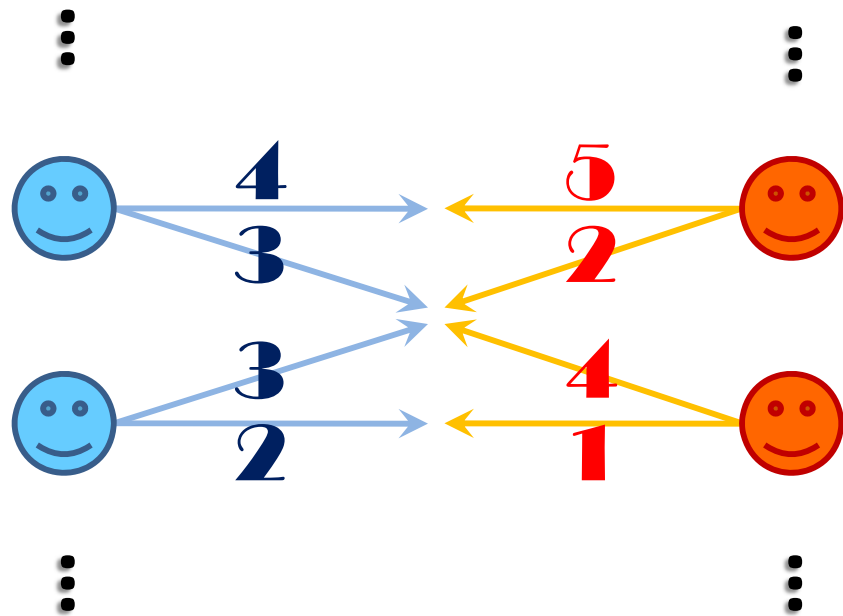
※男女が同数でない場合は、完全マッチング
(perfect matching) は存在しないので、最大マッチン
グ (maximum matching) を求めます。

安定結婚問題

- n 人の男性の集合と、 m 人の女性の集合が存在し、各人は異性全員の**選好順序**をもっている。このとき、**安定なマッチング**を見つけない。



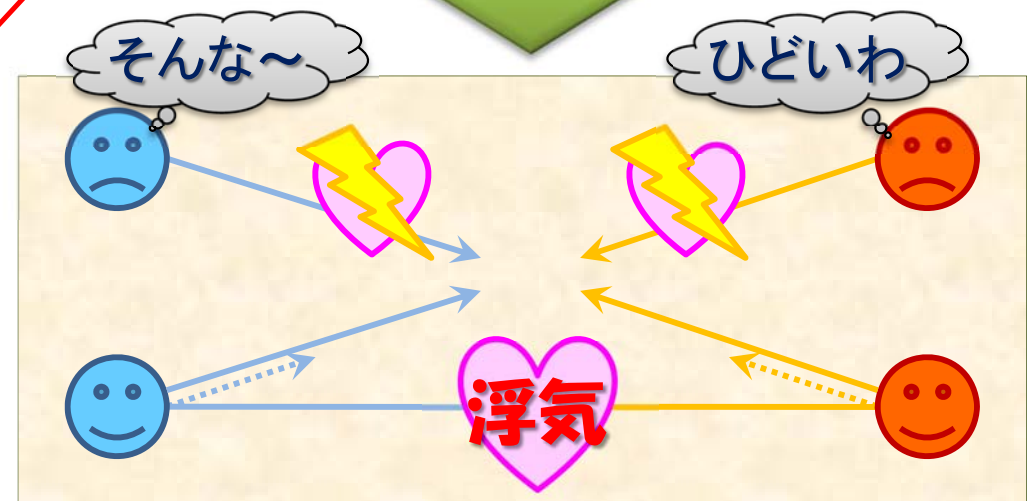
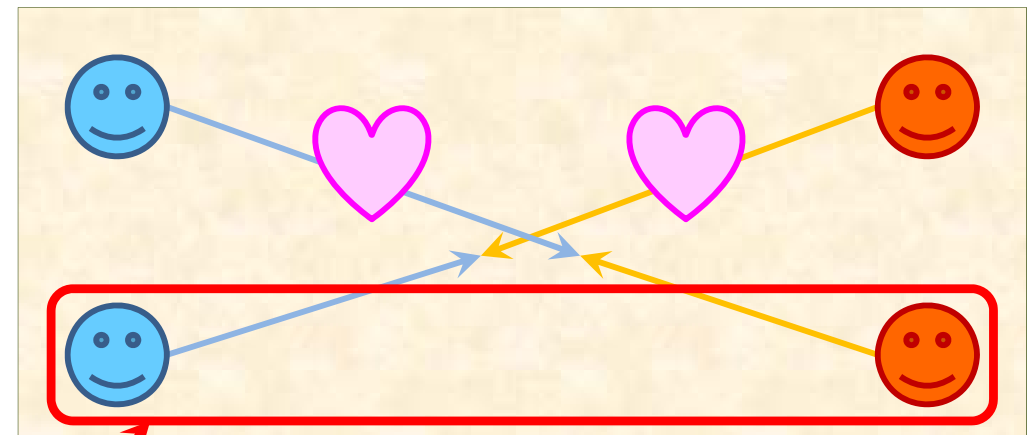
浮気する(不安定な)カップルとは？



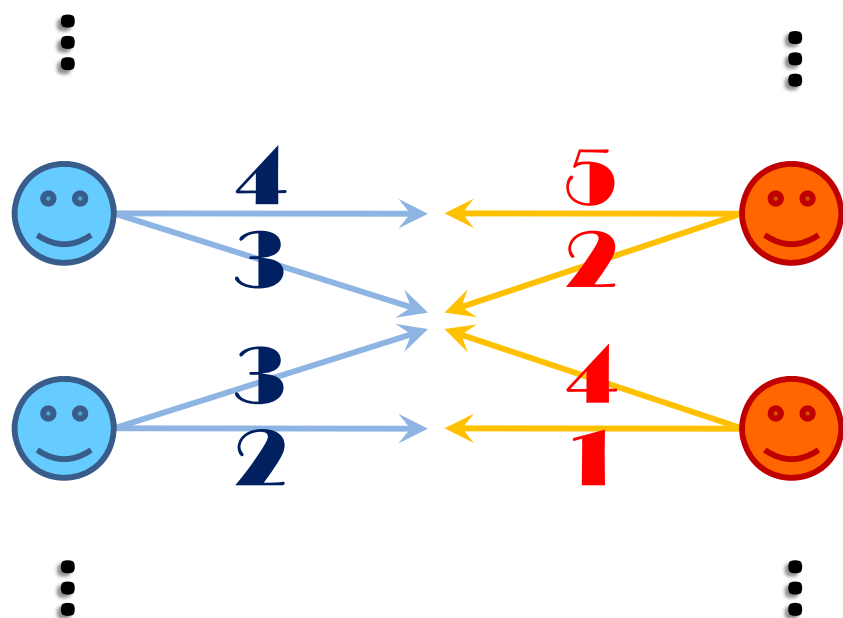
このマッチングは不安定！
なぜなら

ブロッキング・ペア
が存在するから！

こんな2組のカップル(マッチング)を
作ってしまったら...



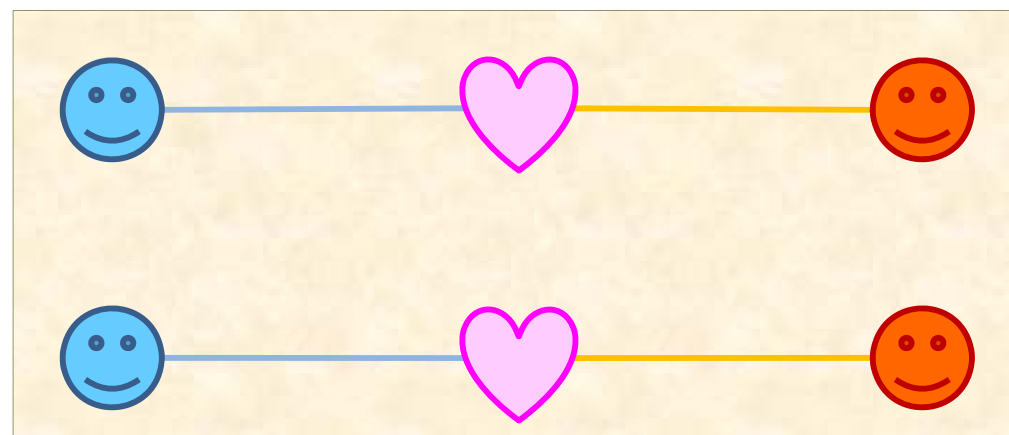
浮気しない(安定な)恋人たち



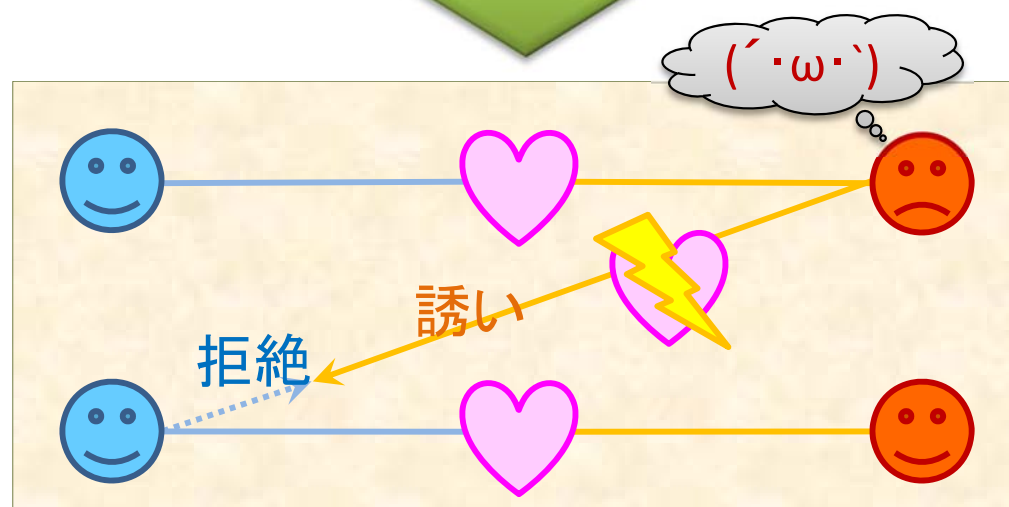
このマッチングは安定！
なぜなら

ブロッキング・ペア
が存在しないから

浮気しない(できない)恋人たち

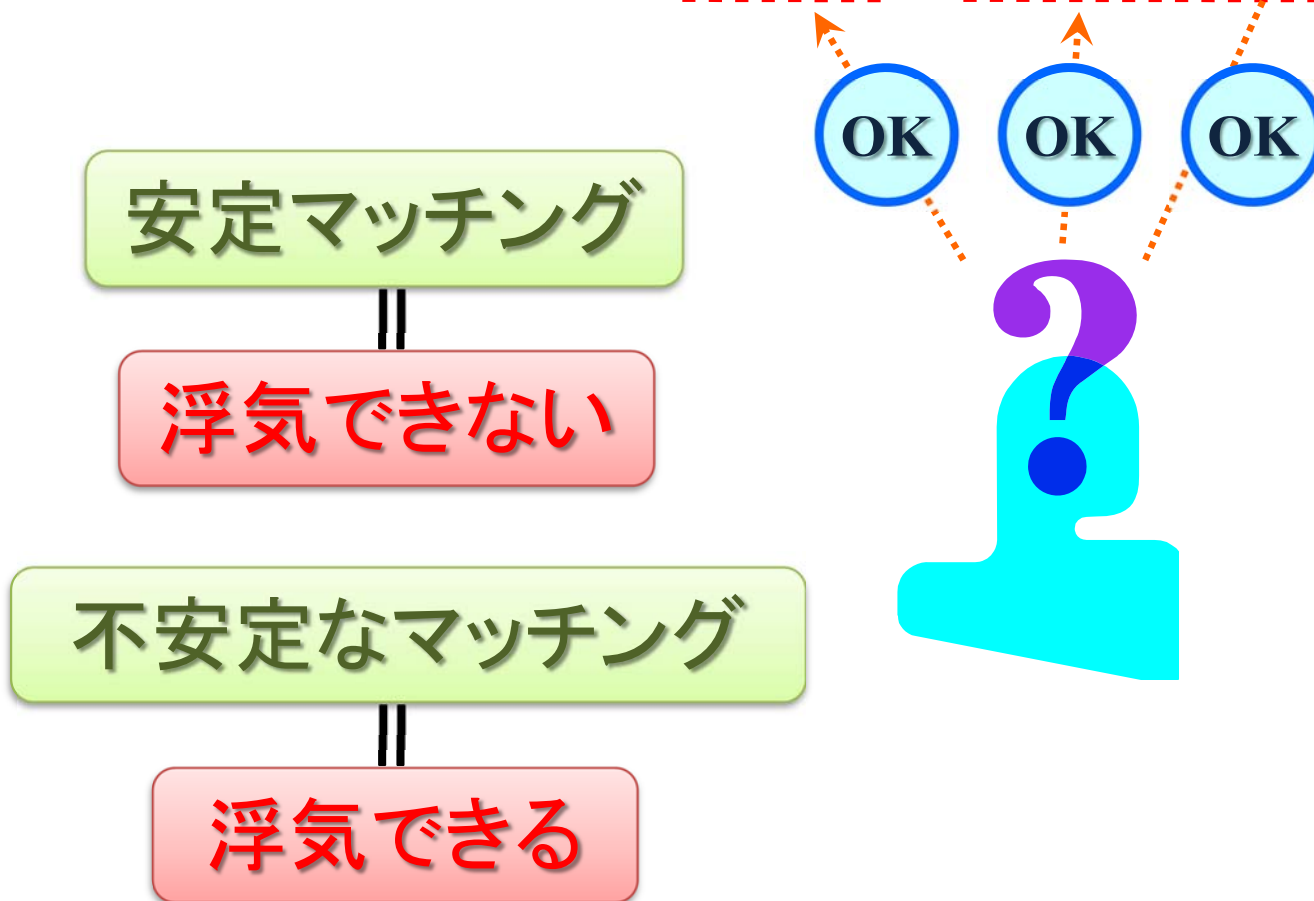


浮気を試みるも...

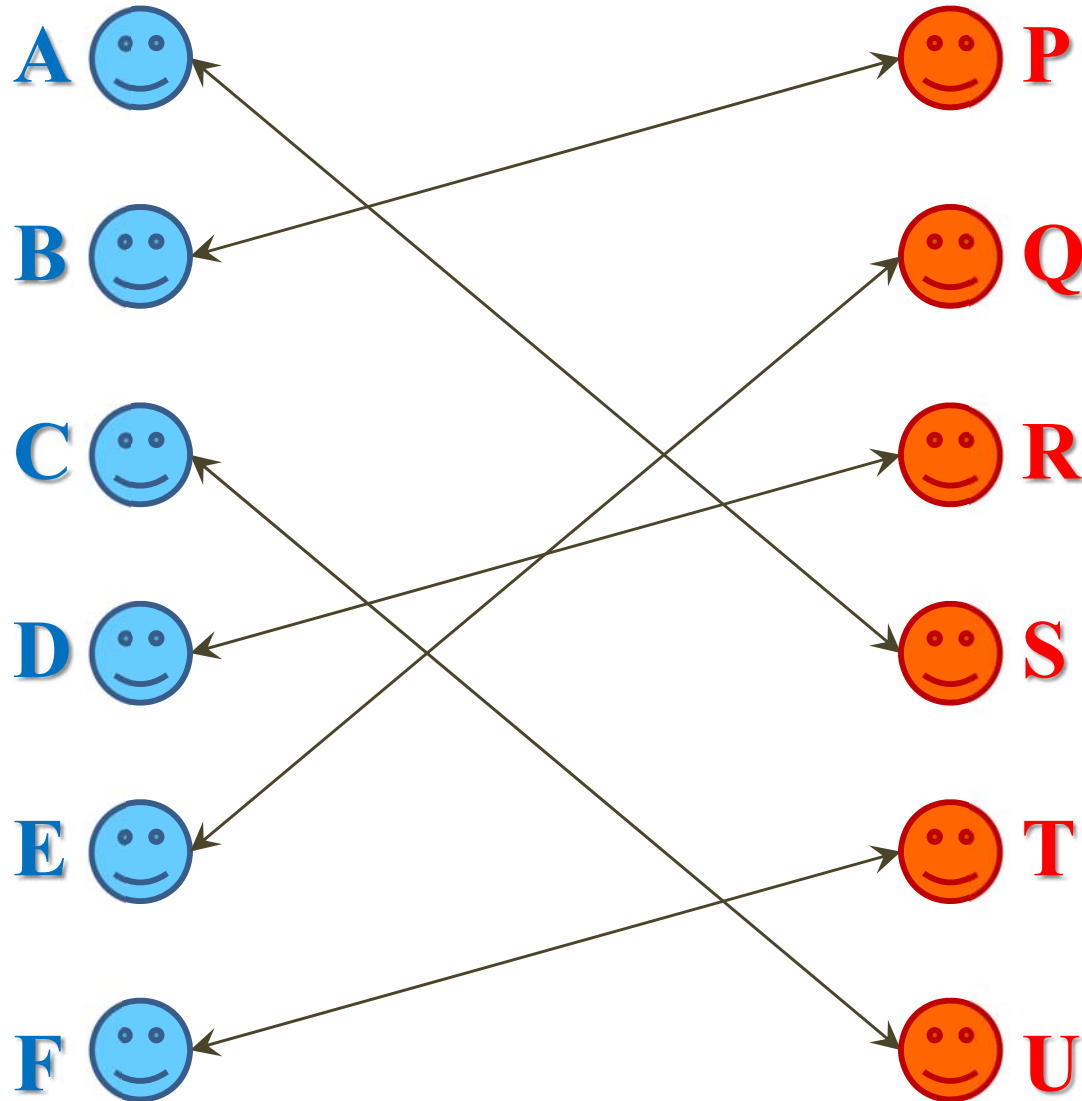


安定結婚問題

- n 人の男性の集合と、 m 人の女性の集合が存在し、各人は異性全員の**選好順序**をもっている。このとき、**安定なマッチング**を見つけない。



安定結婚問題(まとめ)



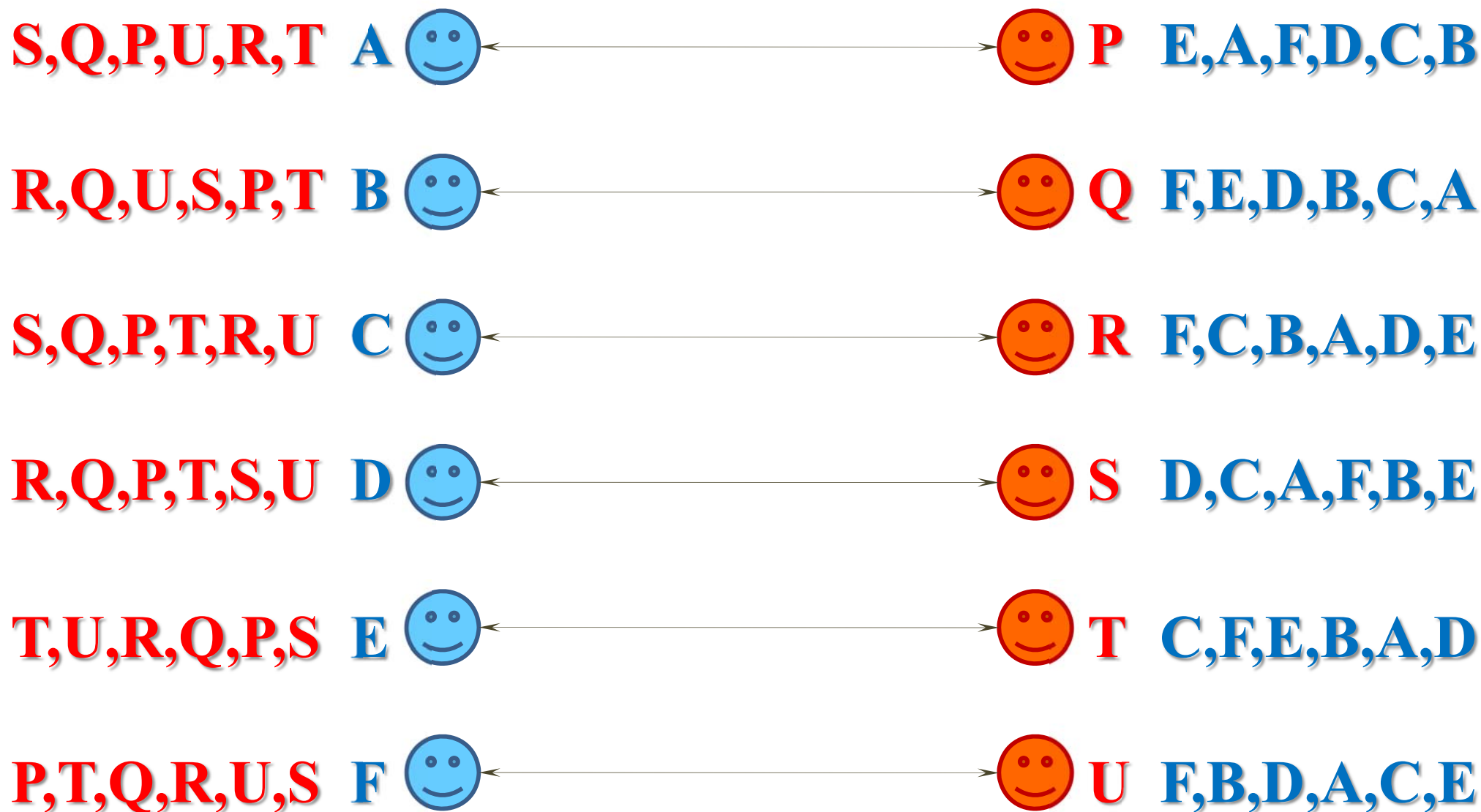
浮気しないカップルをつくる(安定結婚問題を解く)ということは,

(ブロッキング・ペアが存在しない) **安定**な完全マッチングを求める

こと

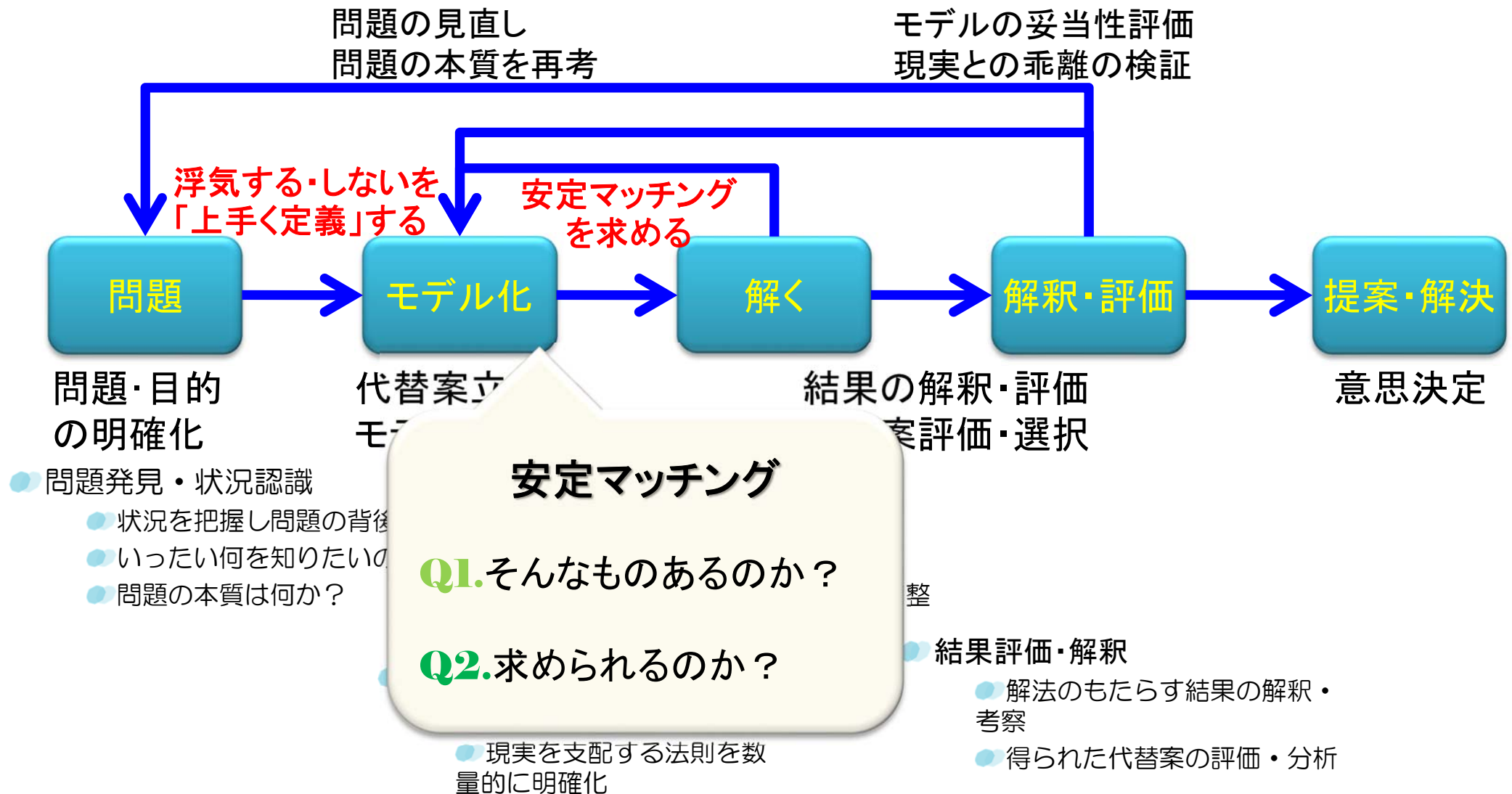
※男女が同数でない場合は, 完全マッチング(perfect matching)は存在しないので, 最大マッチング(maximum matching)を求めます.

問題：このマッチングは安定？

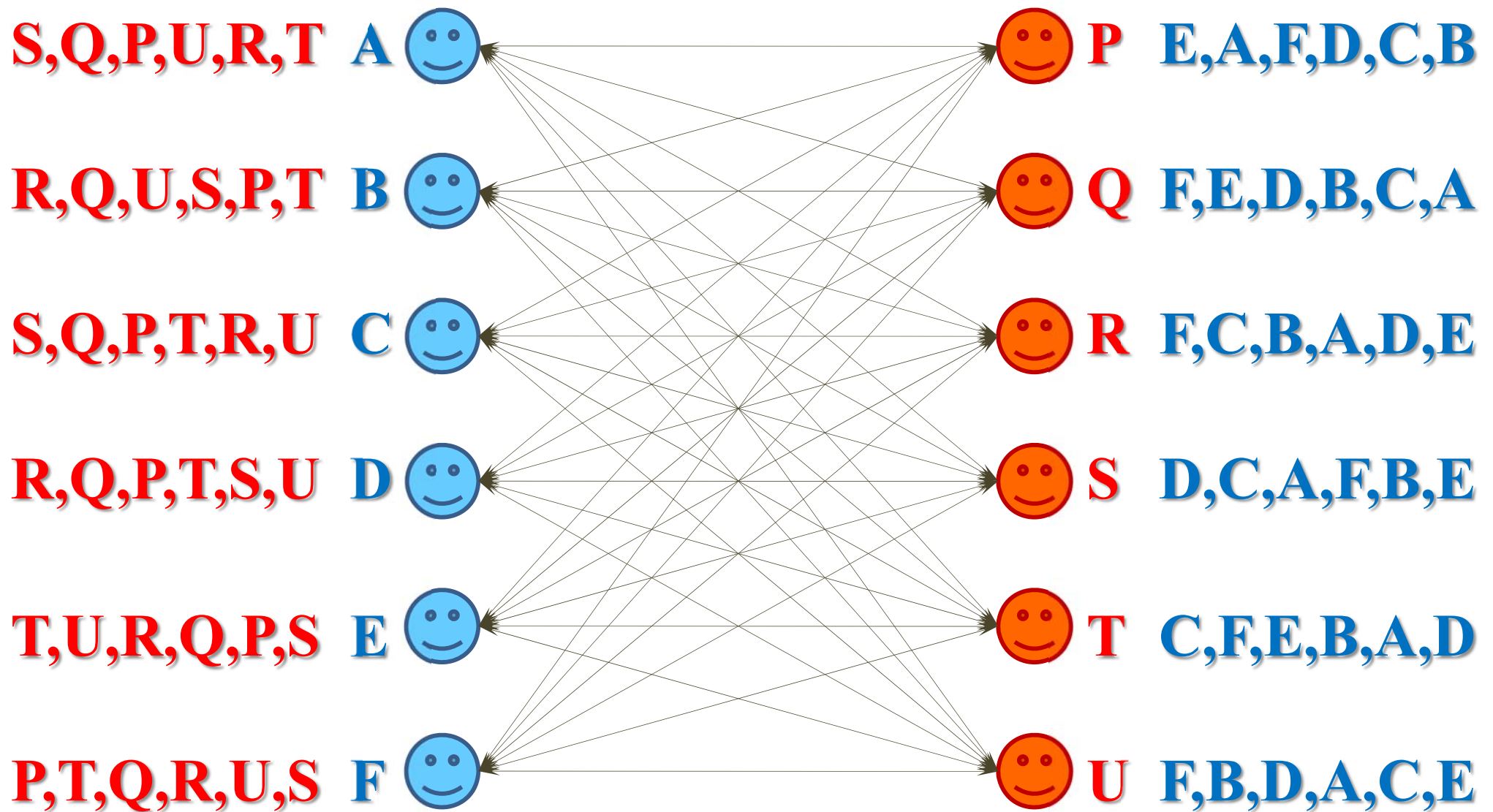


問題解決

「問題の把握」から「意思決定」までの流れ

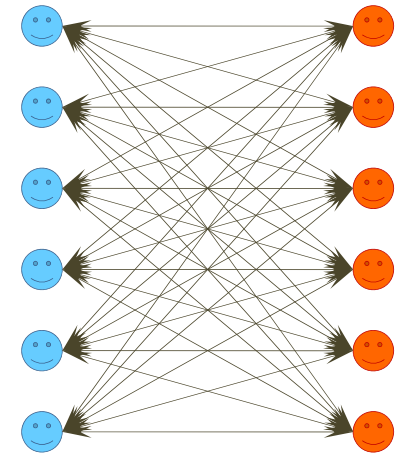


演習：やってみよう



完全マッチングは全部で幾つ？

男女各人数	完全マッチング数
6	720
10	3,628,800
20	2.4×10^{18}
30	2.7×10^{32}
40	8.2×10^{47}
50	3.0×10^{64}
100	9.3×10^{157}
200	#NUM!



※調べた最初の1つが安定解ならそれで計算終了だが、最悪、一番最後まで見つからないかもしれない。また、そもそも安定解など存在しないかもしれないので、その場合は全部調べなければならない



完全マッチングは全部で幾つ？

完全マッチングが膨大にあるとは言っても、今のコンピュータは
かなりの速さで計算できるんでしょ？ だから大丈夫だよね！

- 代表的なCPU, Game機, super computer の浮動小数点演算回数

- Intel Core i7(3.2GHz) : **51.2GFLOPS**

...1秒間に約**512億**回

- PS3 : **218GFLOPS**

...1秒間に約**2180億**回

- PS4 : **1.84TFLOPS**

...1秒間に約**1兆8400億**回

- 京 : **10.51PFLOPS**

...1秒間に約**1京510兆**回

(※2011年6月, 11月 **世界最速!** by Top500.org)

(※2012年6月=2位, 11月=3位, 2013年6月=4位, 11月=4位)

[Wikipedia「FLOPS」より]

2013/5/1の情報

※**FLOPS** = **F**loating-point **O**perations **P**er **S**econd

完全マッチングを一つ見つけるのに、男(女)
の人数(完全マッチング数)の浮動小数点演
算でできると仮定しよう

例えば、 $n=6$ (男6人, 女6人)のときは、6回の演算で計算可
と仮定するということ

K(キロ) $\simeq \times 10^3$ = 千倍

M(メガ) $\simeq \times 10^6$ = 百万倍

G(ギガ) $\simeq \times 10^9$ = 10億倍

T(テラ) $\simeq \times 10^{12}$ = 1兆倍

P(ペタ) $\simeq \times 10^{15}$ = 千兆倍

E(エクサ) $\simeq \times 10^{18}$ = 百京倍

完全マッチングは全部で幾つ？

51.2GFLOPS

1.84TFLOPS

10.51PFLOPS

人数	pm数	Core i7	PS4	京
6	720	0.00000001 秒	0.00000000 秒	0.00000000 秒
10	3,628,800	0.0007088 秒	0.0000197 秒	0.00000000 秒
20	2.4×10^{18}	30 年	306 日	1.3 時間
30	2.7×10^{32}	357,129 宙齡	9,938 宙齡	1.7 宙齡
40	8.2×10^{47}	$1.5E+21$ 宙齡	$4.1E+19$ 宙齡	$7.1E+15$ 宙齡
50	3.0×10^{64}	$6.8E+37$ 宙齡	$1.9E+36$ 宙齡	$3.3E+32$ 宙齡
100	9.3×10^{157}	$4.2E+131$ 宙齡	$1.2E+130$ 宙齡	$2.0E+126$ 宙齡
200	#NUM!	#NUM!	#NUM!	#NUM!

圧倒的な計算力をもつコンピュータですら、**全列挙(しらみつぶし)**では答えを求めることは期待出来ない

1宙齡 = 138億年



補足：スパコンの性能

- Top500 (行列**演算**: 連立一次方程式を解く速度を評価)

- 京: **10.51PFLOPS** ...1秒間に**1京510兆**回

- 2011年6月 1位
- 2011年11月 1位
- 2012年6月 2位
- 2012年11月 3位
- 2013年6月 4位
- 2013年11月 4位
- 2014年6月 4位
- 2014年11月 4位
- 2015年6月 4位

他に**Green500**なども
(エネルギー消費効率の良さを競う **FLOPS per Watt**)
2015年6月上位3機は日本
1位. 菖蒲, 2位. 青睡蓮, 3位. 睡蓮

※FLOPS = **FL**loating-point **O**perations **P**er **S**econd

※TEPS = **T**raversed **E**dges **P**er **S**econd

- Graph500 (大規模**グラフ**解析の性能を評価)

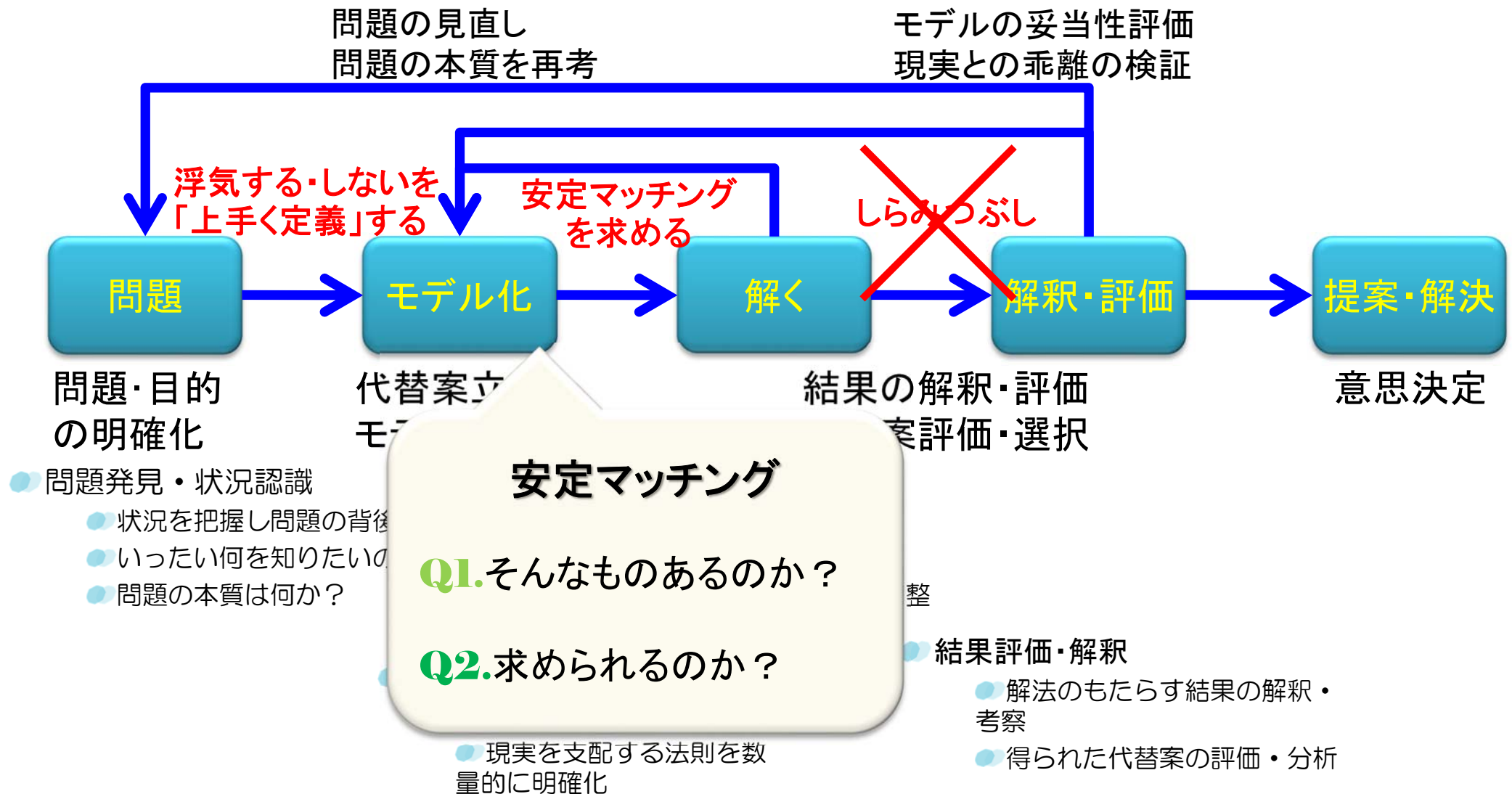
- 京: **38,621GTEPS** ...1秒間に**38兆6210億**個

- 2014年6月 1位
- 2014年11月 2位
- 2015年6月 1位 (約1兆個の点, 約16兆個の枝からなるグラフの幅優先探索を0.45秒で処理)

✓ 計算速度
✓ アルゴリズム
✓ プログラム
などの**総合力**の競争

問題解決

「問題の把握」から「意思決定」までの流れ



ではどうする？

- 素朴で素直な方法〔列挙法〕
 - 全ての完全マッチングをしらみつぶしに調べて、安定解を探す

時間が
掛かり過
ぎる！



全ての完全マッチングを
しらみつぶしに調べずに、
安定解を、現実的時間で
見つける方法があるか？

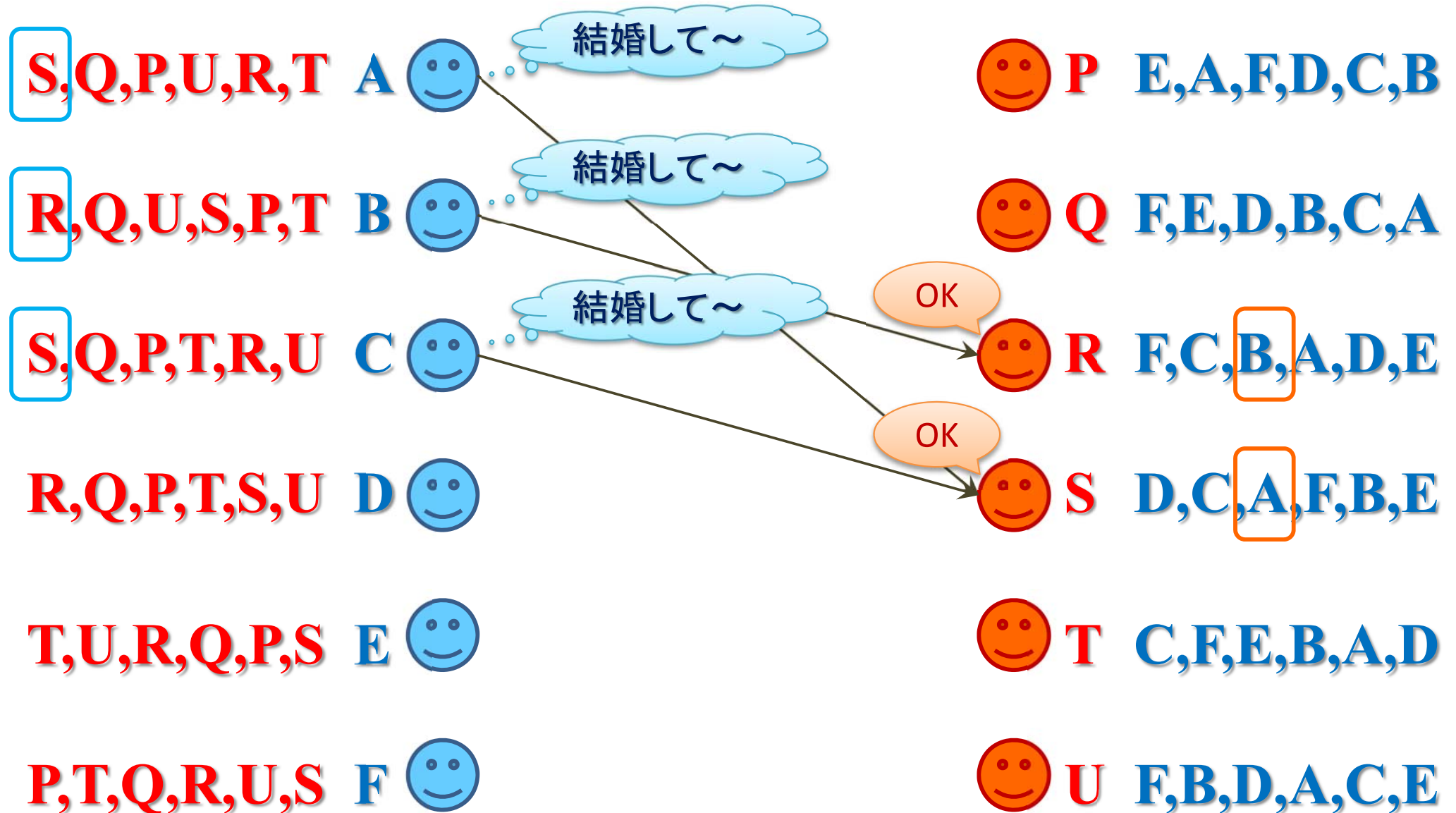
Gale-Shapley
Algorithm

人間の創造
的な仕事！

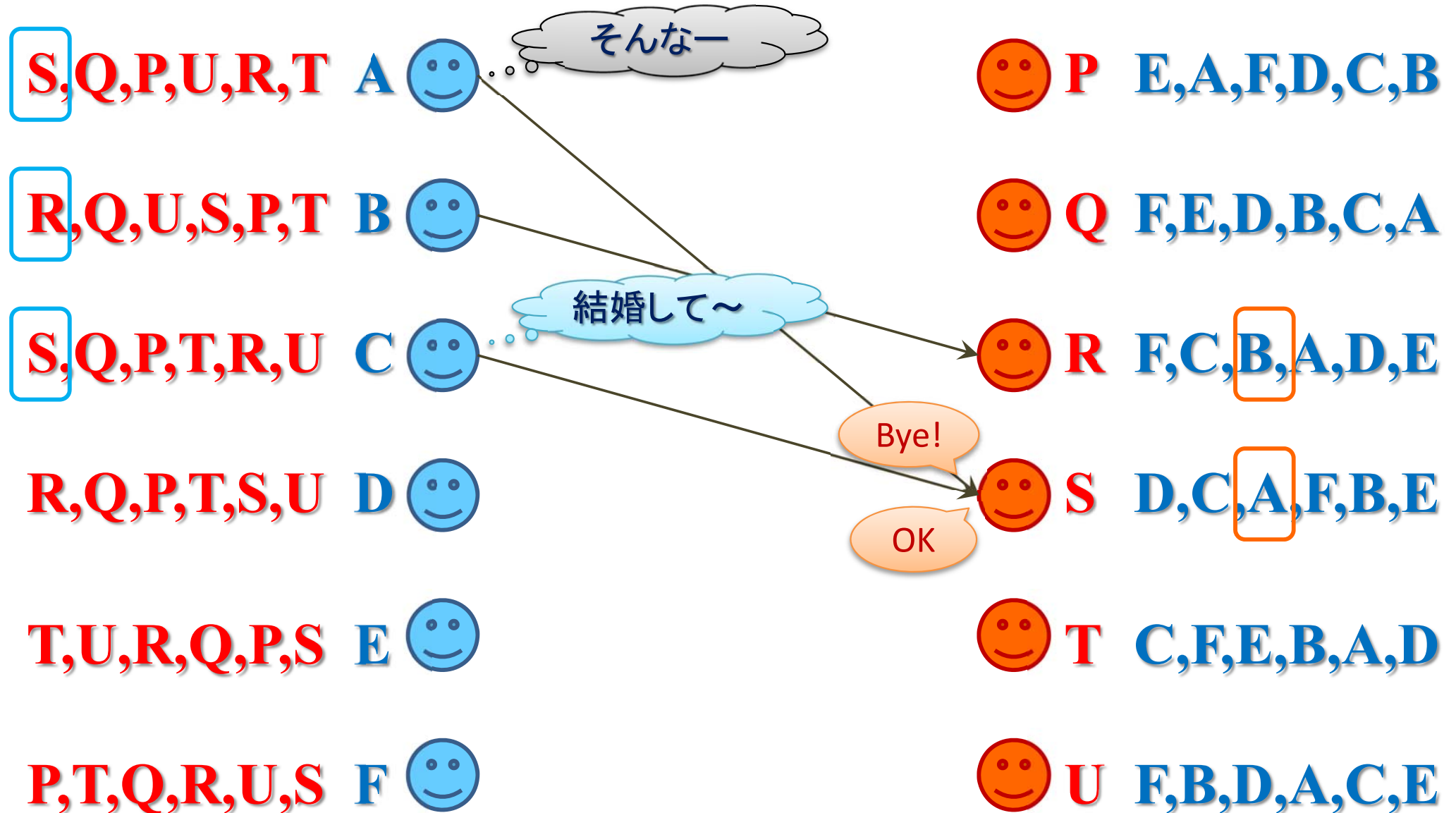
安定結婚問題を解く

Gale-Shapley アルゴリズム

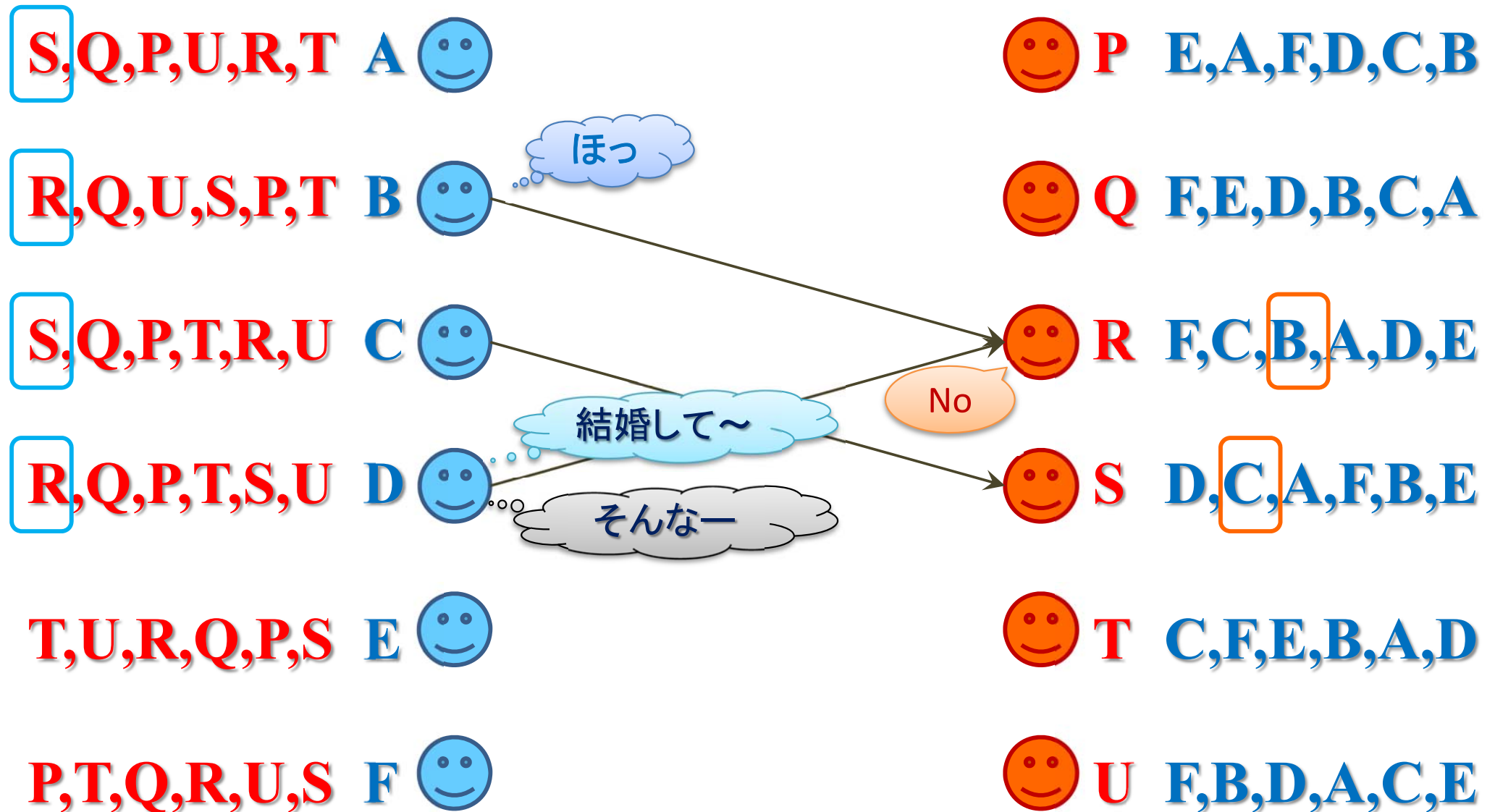
Gale-Shapley アルゴリズム



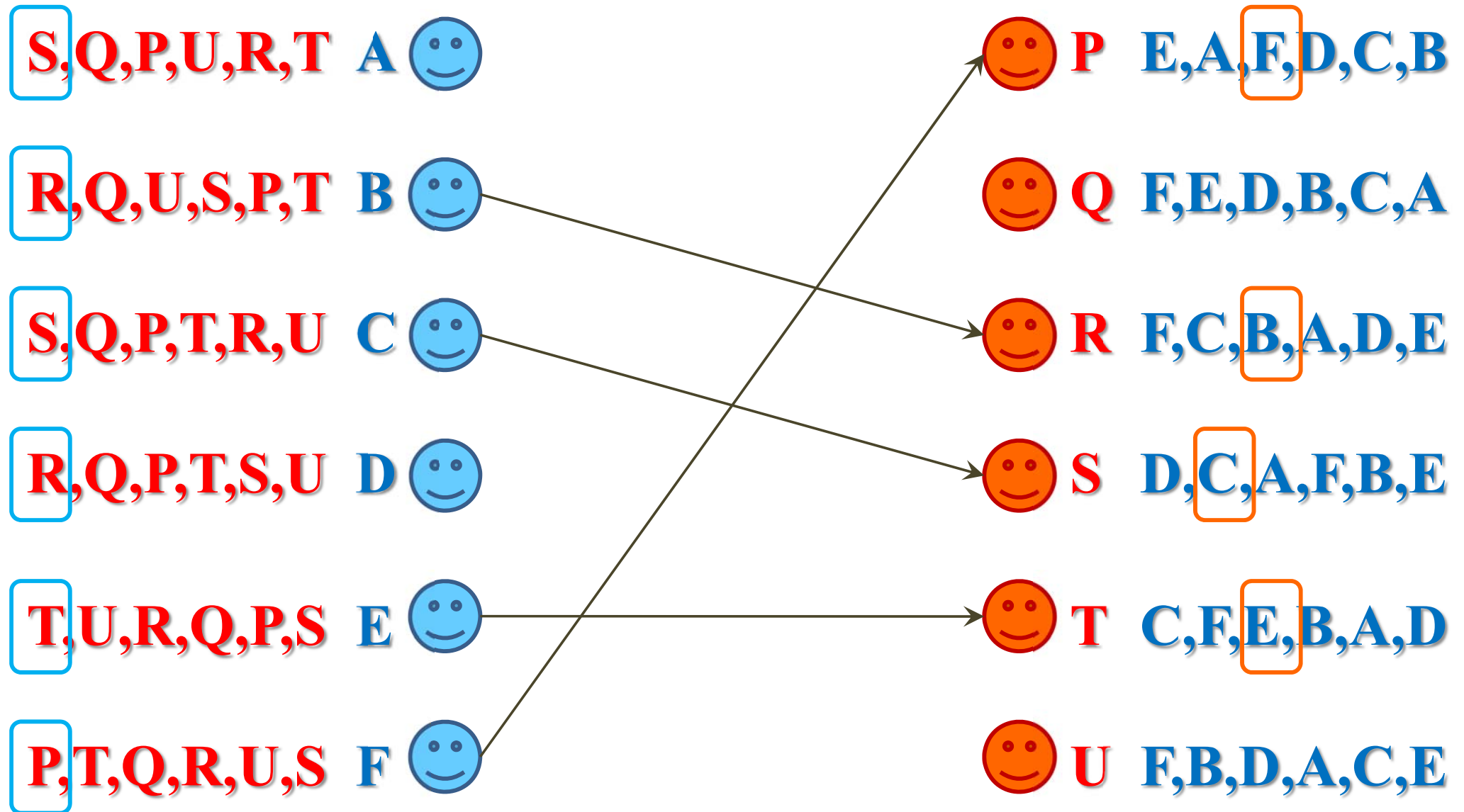
Gale-Shapley アルゴリズム



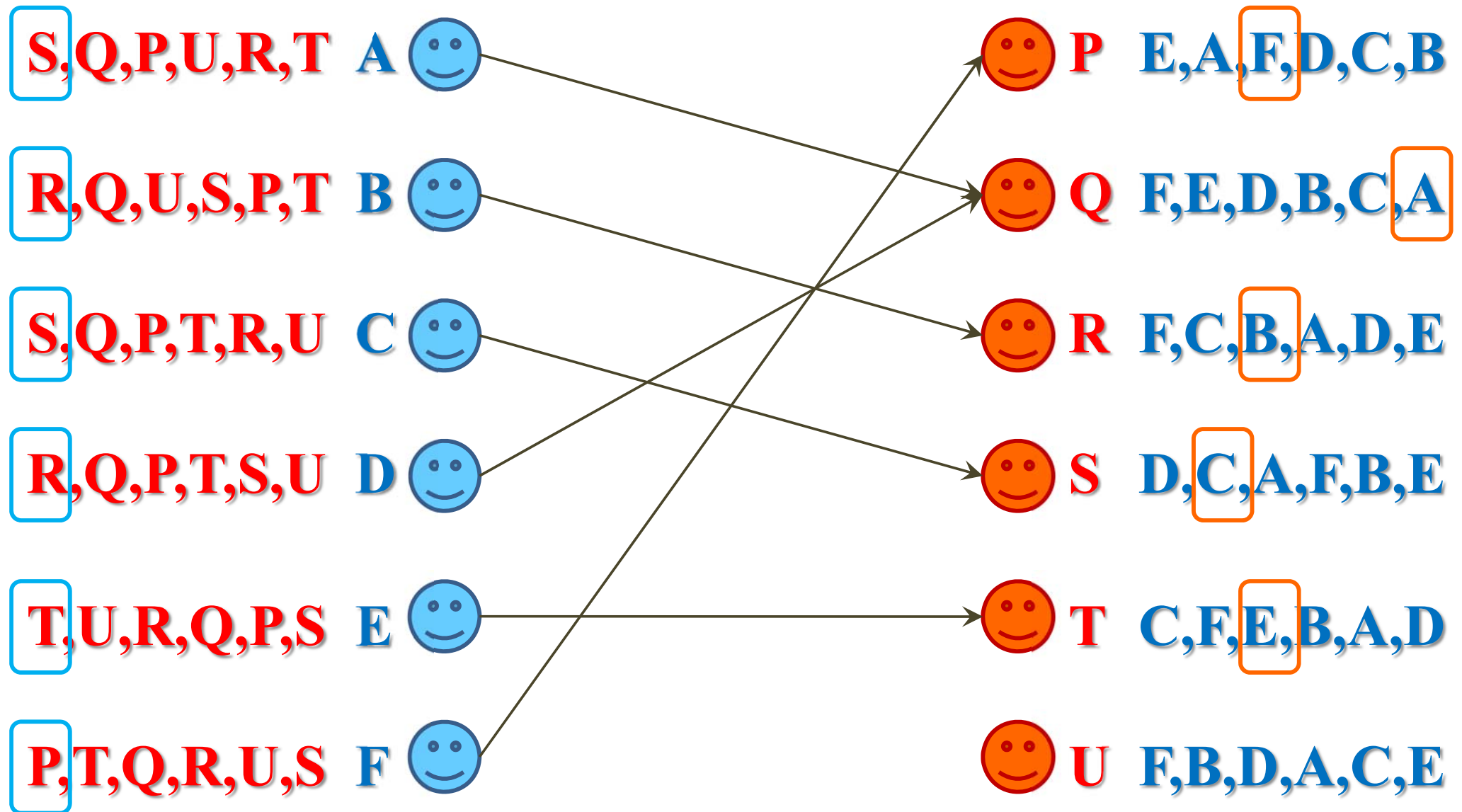
Gale-Shapley アルゴリズム



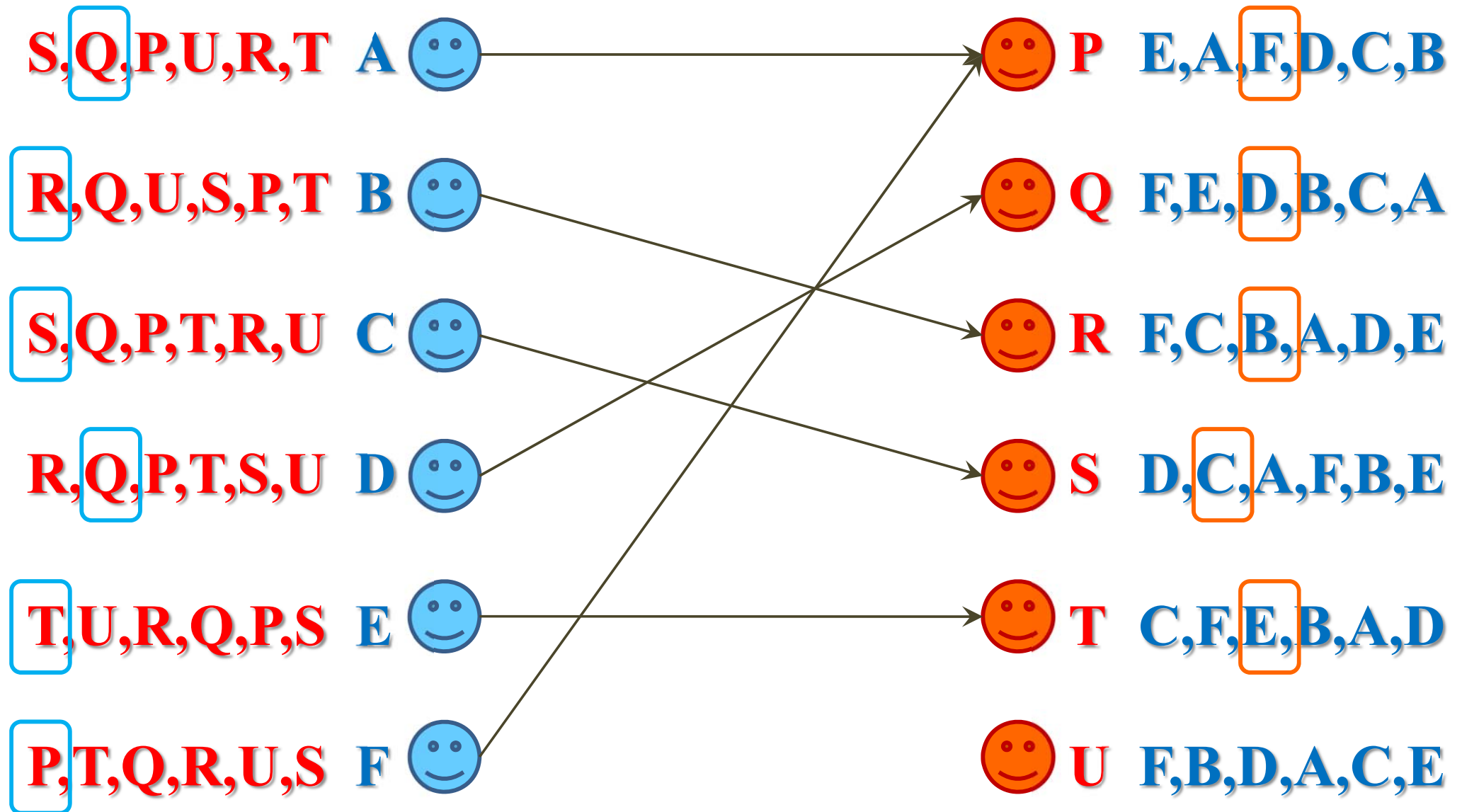
Gale-Shapley アルゴリズム



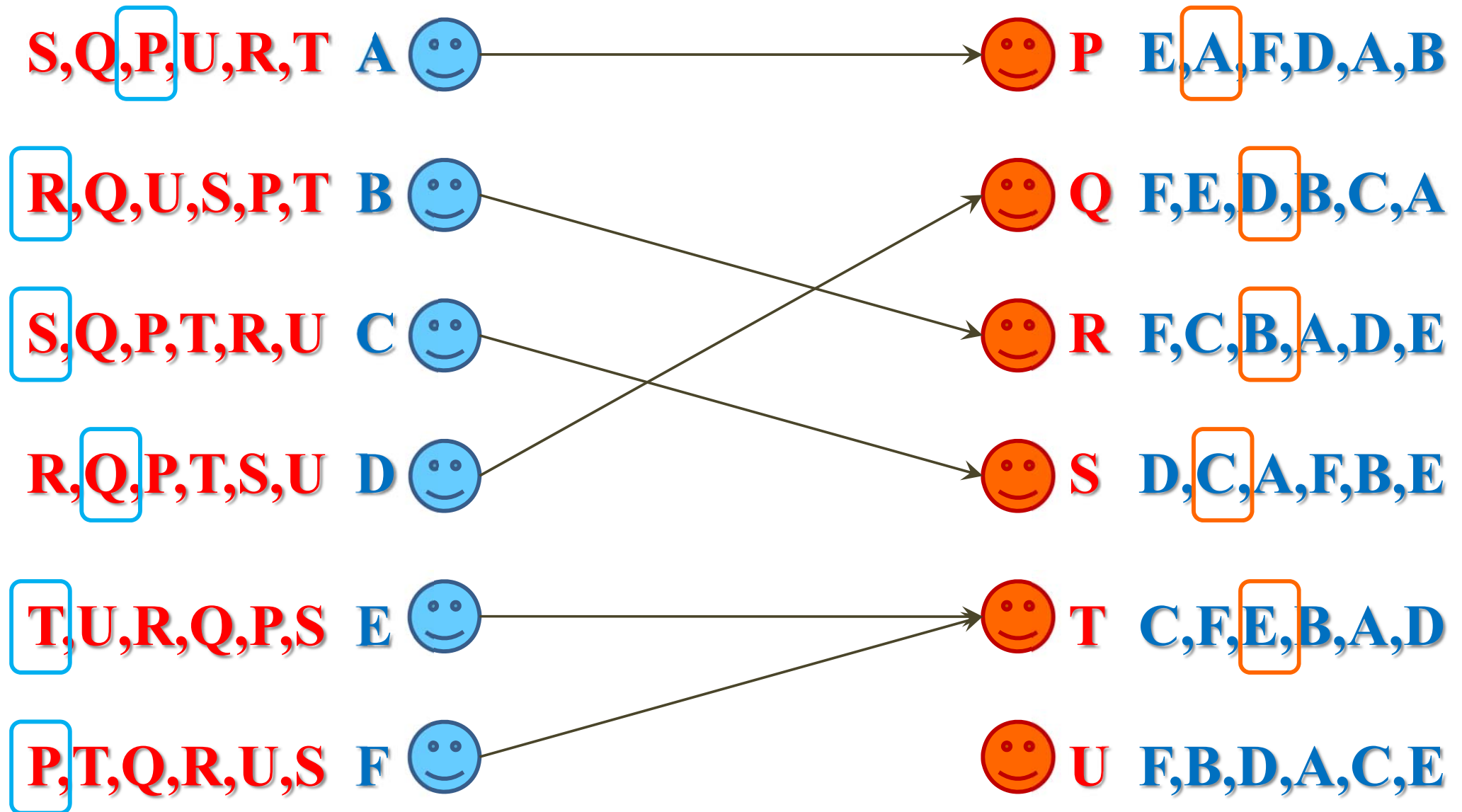
Gale-Shapley アルゴリズム



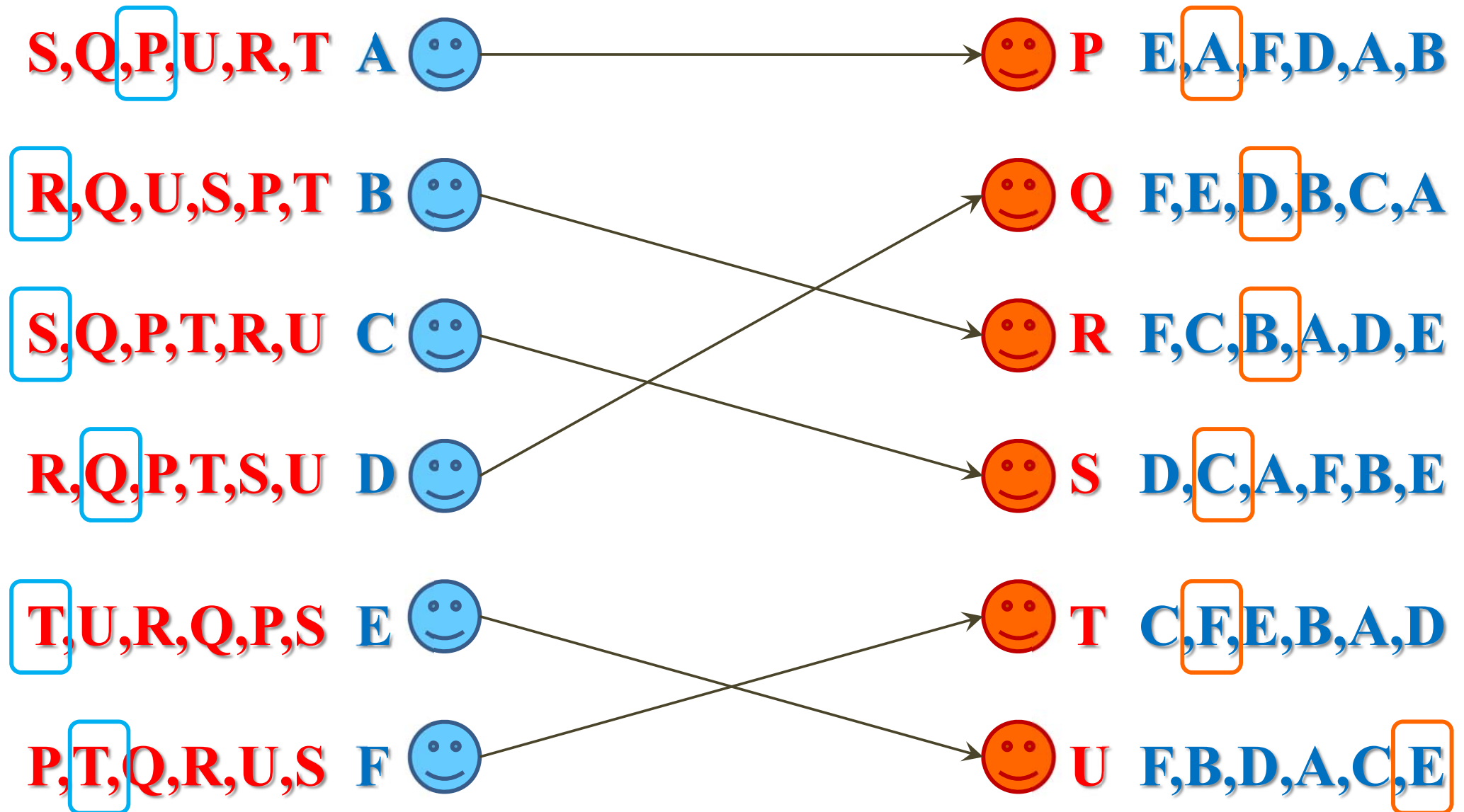
Gale-Shapley アルゴリズム



Gale-Shapley アルゴリズム



Gale-Shapley アルゴリズム

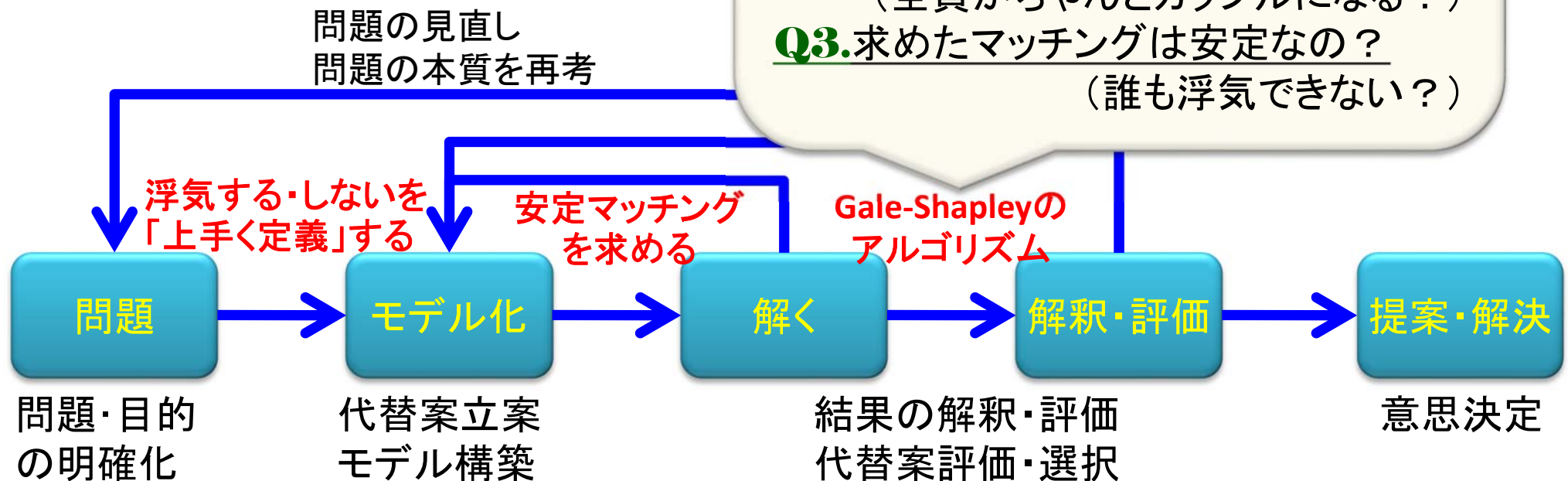


問題解決

「問題の把握」から「意思決定」まで

アルゴリズムの評価

- Q1.** アルゴリズムはちゃんと終わる？
(無限に続くことはない？)
- Q2.** 完全マッチングを求めたのか？
(全員がちゃんとカップルになる？)
- Q3.** 求めたマッチングは安定なの？
(誰も浮気できない？)



問題発見・状況認識

- 状況を把握し問題の背後にある本質を追究
- いったい何を知りたいのか？
- 問題の本質は何か？

答えを導く

- 解法選択
- 解法構築
- パラメータ調整

推論・モデル作成

- 推論に基づきモデル作成
- 現実を支配する法則を数
量的に明確化

結果評価・解釈

- 解法のもたらす結果の解釈・
考察
- 得られた代替案の評価・分析

評価 : Gale-Shapley Alg. の解の評価

- **定理** : 与えられた安定結婚問題における任意の選好順位に対し, Gale-Shapley アルゴリズムは 安定マッチング を導き終了する.



A1. きちんと終わるよ！

A2. 完全マッチングを求めるよ！

A3. 安定だよ！

- **系** : 安定結婚問題におけるどのような選好順位に対しても, 少なくとも一つの安定マッチングが存在する.

評価 : Gale-Shapley Alg. って速いの？



- 男(女)の数を n とすると, 大雑把な見積もりで,

$$O(n^2)$$

多項式オーダー

コンピュータに計算させてみよう！

簡単のため $10n^2$ の浮動小数点演算回数で計算できると仮定

人数	pm数	京 & しらみつぶし	Core i7 & GS Alg
6	720	0.00000000秒	0.00000000秒
10	3,628,800	0.00000000秒	0.00000000秒
20	2.4×10^{18}	1.3時間	0.00000001秒
30	2.7×10^{32}	1.7宙齡	0.00000002秒
40	8.2×10^{47}	$7.1\text{E}+15$ 宙齡	0.00000003秒
50	3.0×10^{64}	$3.3\text{E}+32$ 宙齡	0.00000005秒
100	9.3×10^{157}	$2.0\text{E}+126$ 宙齡	0.00000020秒
200	#NUM!	#NUM!	0.00000078秒
1000	#NUM!	#NUM!	0.0001953秒
10000	#NUM!	#NUM!	0.0195313秒
100000	#NUM!	#NUM!	1.9531250秒
1000000	#NUM!	#NUM!	195.3125000秒

世界最速 SuperComp
+ **力技** (しょぼい方法)

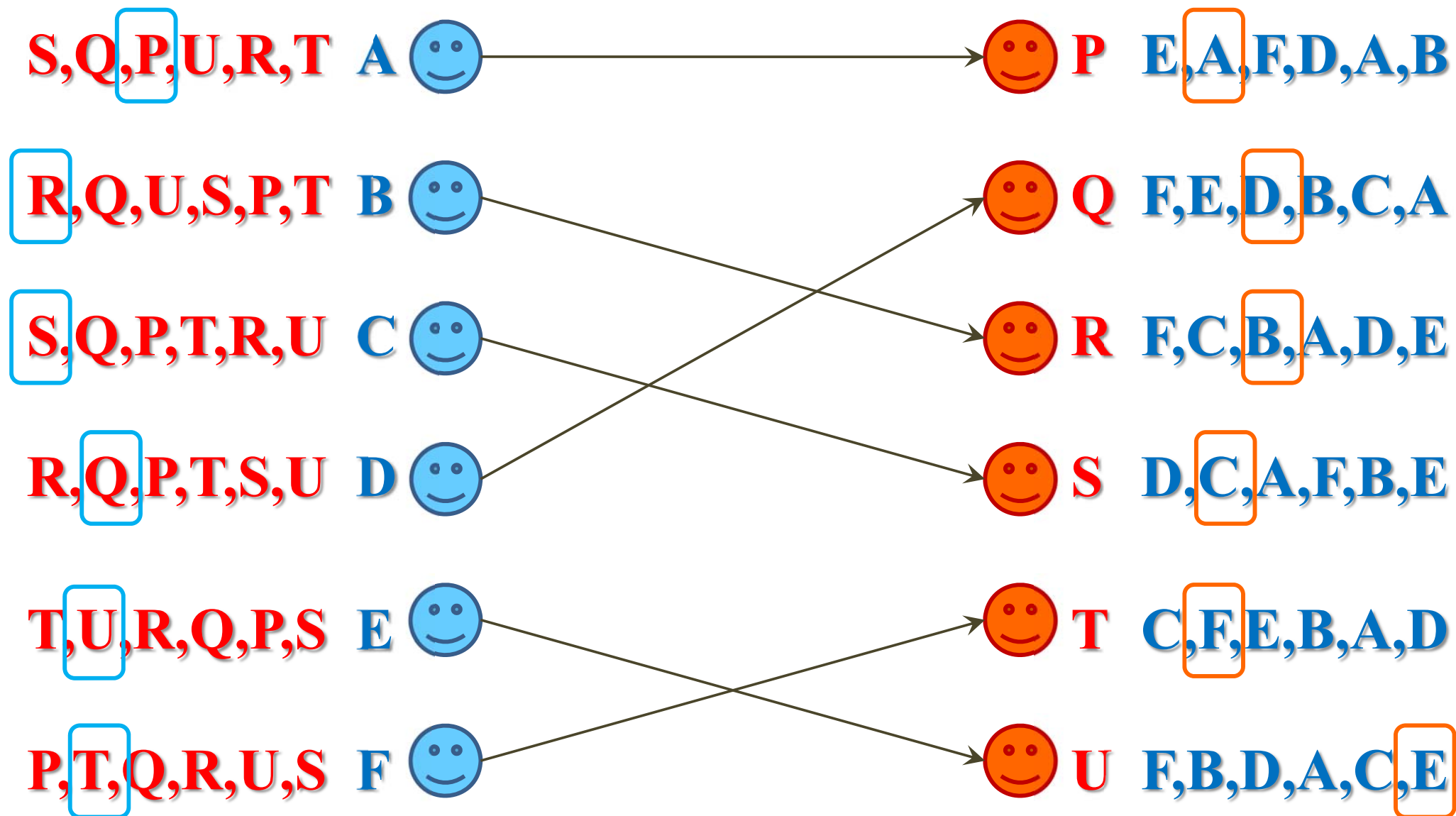


そこらのPC
+ **人間の知恵**

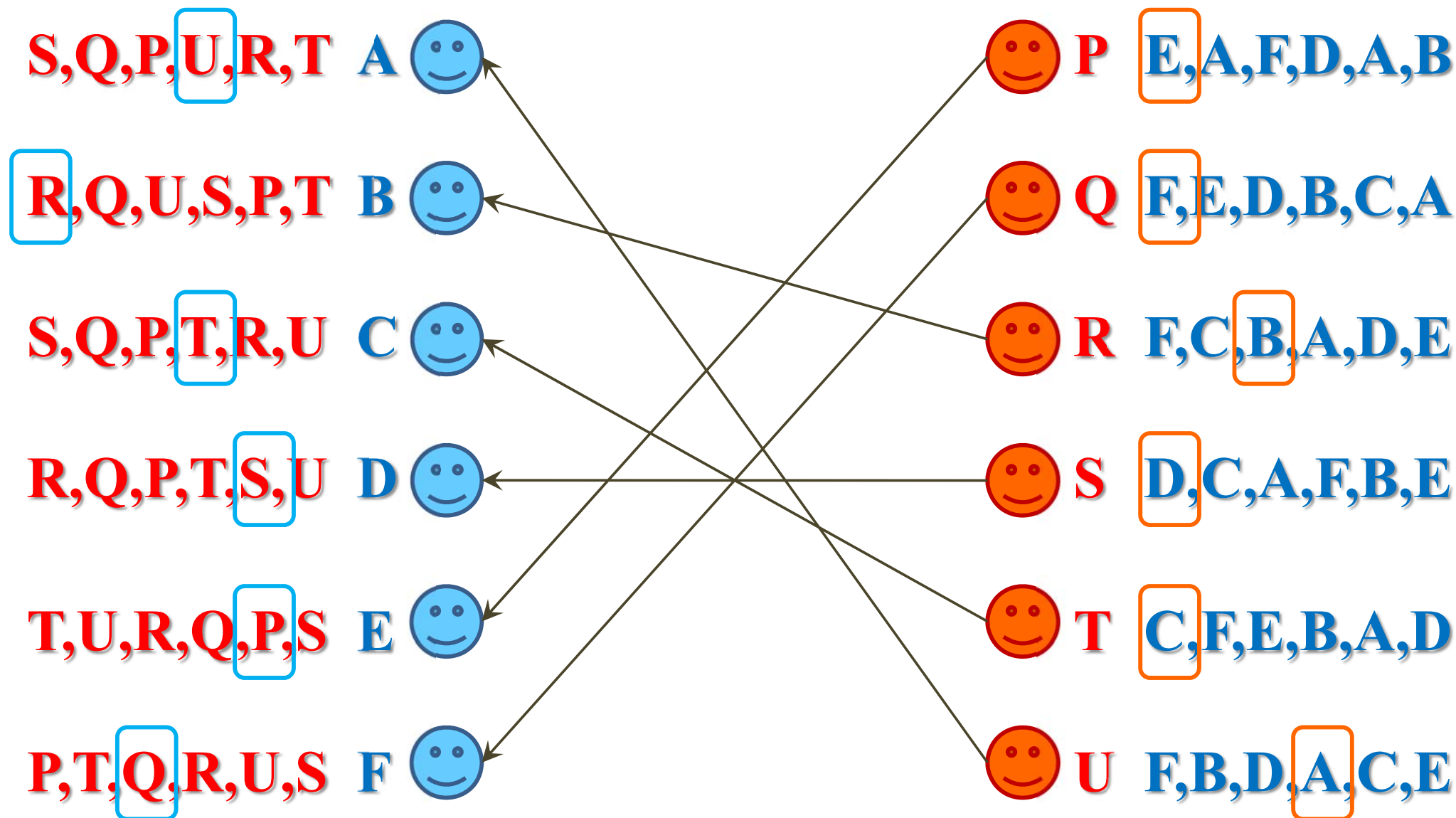
評価 : Gale-Shapley Alg. の解の評価2

- **定理** : 男性側のプロポーズの順番に関係なく, Gale-Shapleyアルゴリズムは, 同一の安定マッチングを導く.
- **系** : 安定結婚問題におけるどのような選好順位に対しても, Gale-Shapleyアルゴリズムは, 男性側からプロポーズすれば男性最良安定マッチングを導く.

男性最良安定マッチング



女性最良安定マッチング



評価: Gale-Shapley Alg. の解の評価3

- 与えられた安定結婚問題について、いくつかの安定マッチングが存在する場合、男性にとってより好ましい安定マッチング、女性にとってより好ましい安定マッチングなど、安定マッチングの**好ましさにある種の順序付け**ができる。

- 定理:** 与えられた安定結婚問題について、
男性最良安定マッチング = 女性最悪安定マッチング
男性最悪安定マッチング = 女性最良安定マッチング
である。

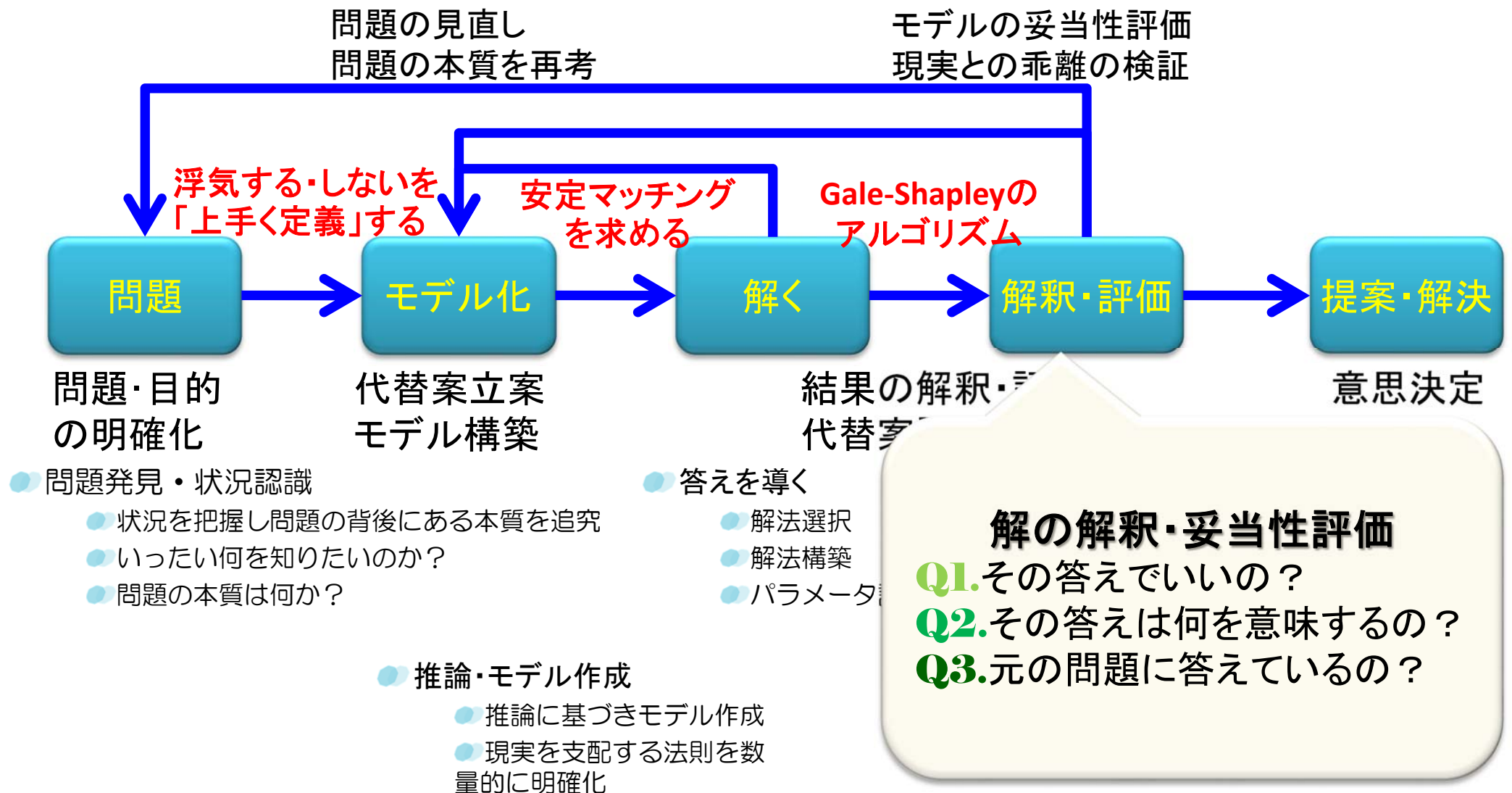


教訓!? 『待ってちゃダメ！

好きになったら自分から告白しなさい』

問題解決

「問題の把握」から「意思決定」までの流れ



もっと知りたい人へ

- OR入門書・啓蒙書

- 久保, 松井「**組合せ最適化『短編集』**」朝倉書店(1999)
- 山本, 久保「**巡回セールスマン問題への招待**」朝倉書店(1997)
- グリッツマン, ブランデンベルク「**最短経路の本**」シュプリンガー(2008)
- 松井, 根本, 宇野「**入門オペレーションズ・リサーチ**」東海大出版(2008)
- W.J.クック「**驚きの数学 巡回セールスマン問題**」青土社(2013)

- さらに詳しい内容を勉強したい人は

- 根本「**安定結婚問題**」(久保, 田村, 松井『応用数理計画ハンドブック』Ch14-2) 朝倉書店(2002)

- 関連する経営学科の授業

- 「**ネットワークモデル分析**」(4セメ)
- 「**最適化モデル分析**」(5セメ)
- 「**意思決定科学**」(6セメ)

etc...

練習:

男性最良安定マッチングを求めよ(プロポーズは上から順に一人ずつ)

選好順

1	2	3	4
P	Q	R	S

A 

P	S	Q	R
---	---	---	---

B 

Q	S	R	P
---	---	---	---

C 

R	Q	P	S
---	---	---	---

D 

選好順

1	2	3	4
B	A	C	D

 P

A	C	B	D
---	---	---	---

 Q

A	D	C	B
---	---	---	---

 R

B	C	D	A
---	---	---	---

 S

Gale-Shapley Algorithm
のプロポーズ順

1	→
2	→
3	→
4	→
5	→
6	→
7	→
8	→
9	→
10	→
11	→
12	→
13	→
14	→

練習: 解答例

男性最良安定マッチングを求めよ(プロポーズは上から順に一人ずつ)

1	2	3	4
P	Q	R	S

P	S	Q	R
----------	----------	----------	----------

Q	S	R	P
----------	----------	----------	----------

R	Q	P	S
----------	----------	----------	----------

選好順

1	2	3	4
B	A	C	D

A	C	B	D
----------	----------	----------	----------

A	D	C	B
----------	----------	----------	----------

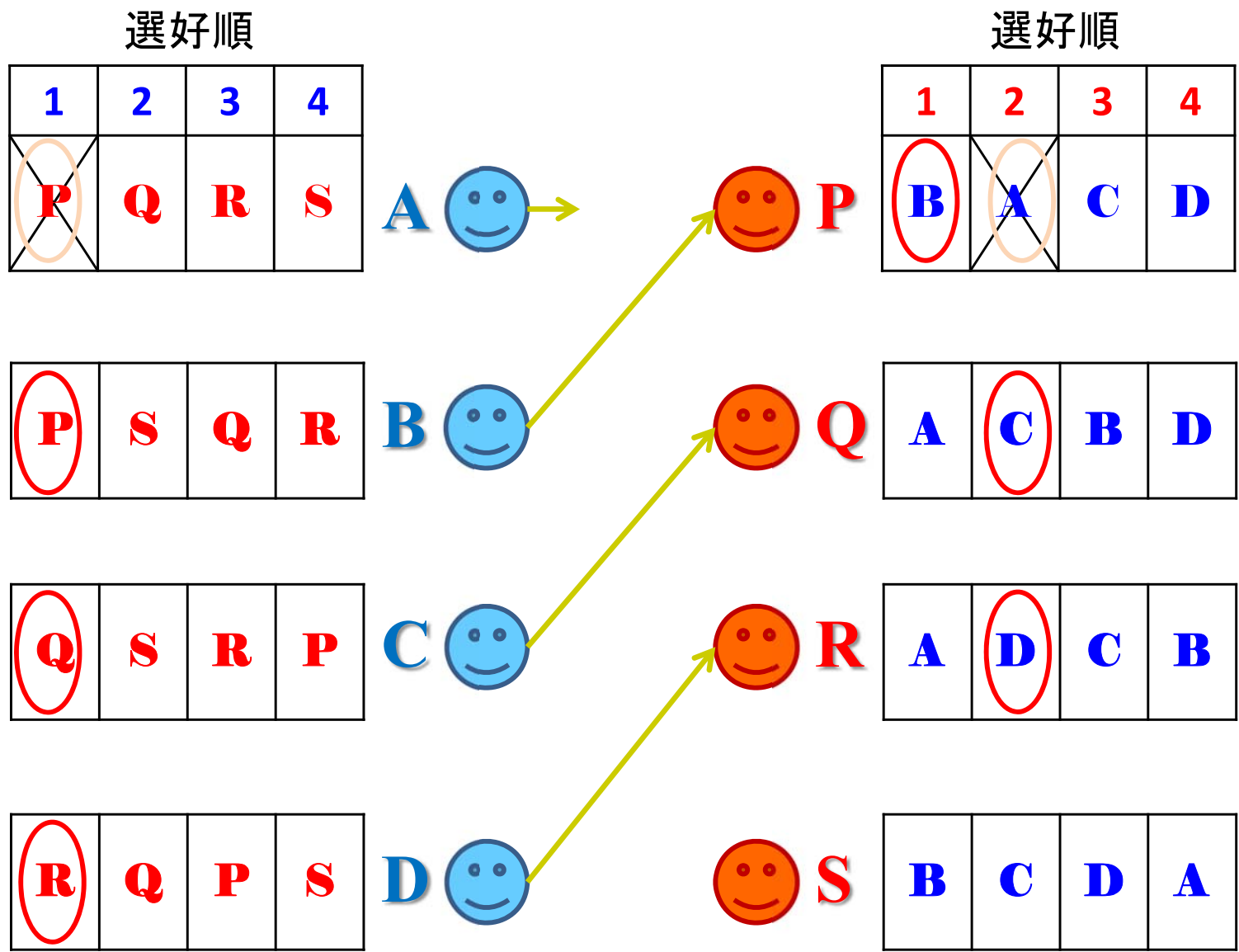
B	C	D	A
----------	----------	----------	----------

Gale-Shapley Algorithm のプロポーズ順

1	A → P
2	B → P
3	→
4	→
5	→
6	→
7	→
8	→
9	→
10	→
11	→
12	→
13	→
14	→

練習: 解答例

男性最良安定マッチングを求めよ(プロポーズは上から順に一人ずつ)

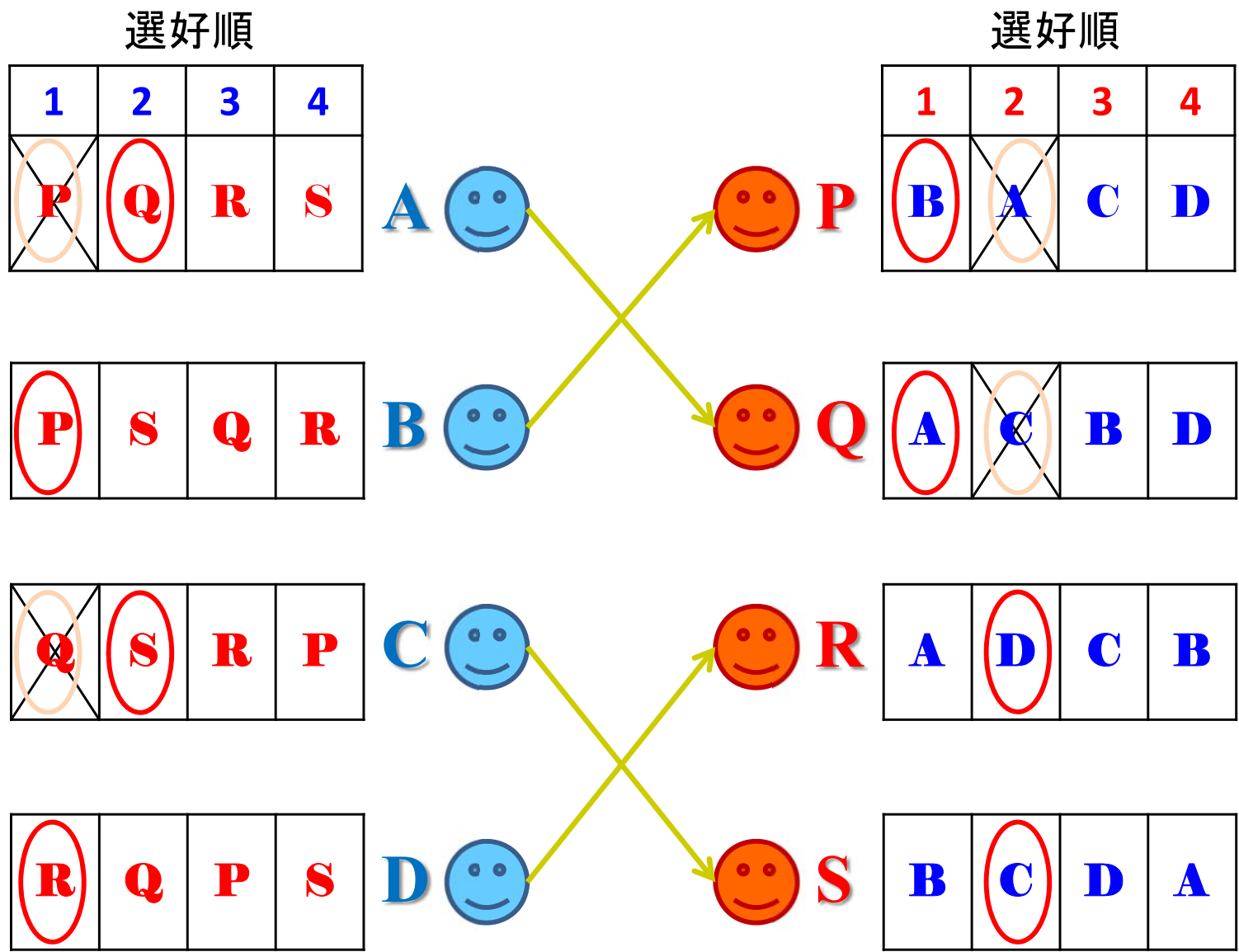


Gale-Shapley Algorithm
のプロポーズ順

1	A → P
2	B → P
3	C → Q
4	D → R
5	→
6	→
7	→
8	→
9	→
10	→
11	→
12	→
13	→
14	→

練習: 解答例

男性最良安定マッチングを求めよ(プロポーズは上から順に一人ずつ)



Gale-Shapley Algorithm
のプロポーズ順

1	A → P
2	B → P
3	C → Q
4	D → R
5	A → Q
6	C → S
7	→
8	→
9	→
10	→
11	→
12	→
13	→
14	→