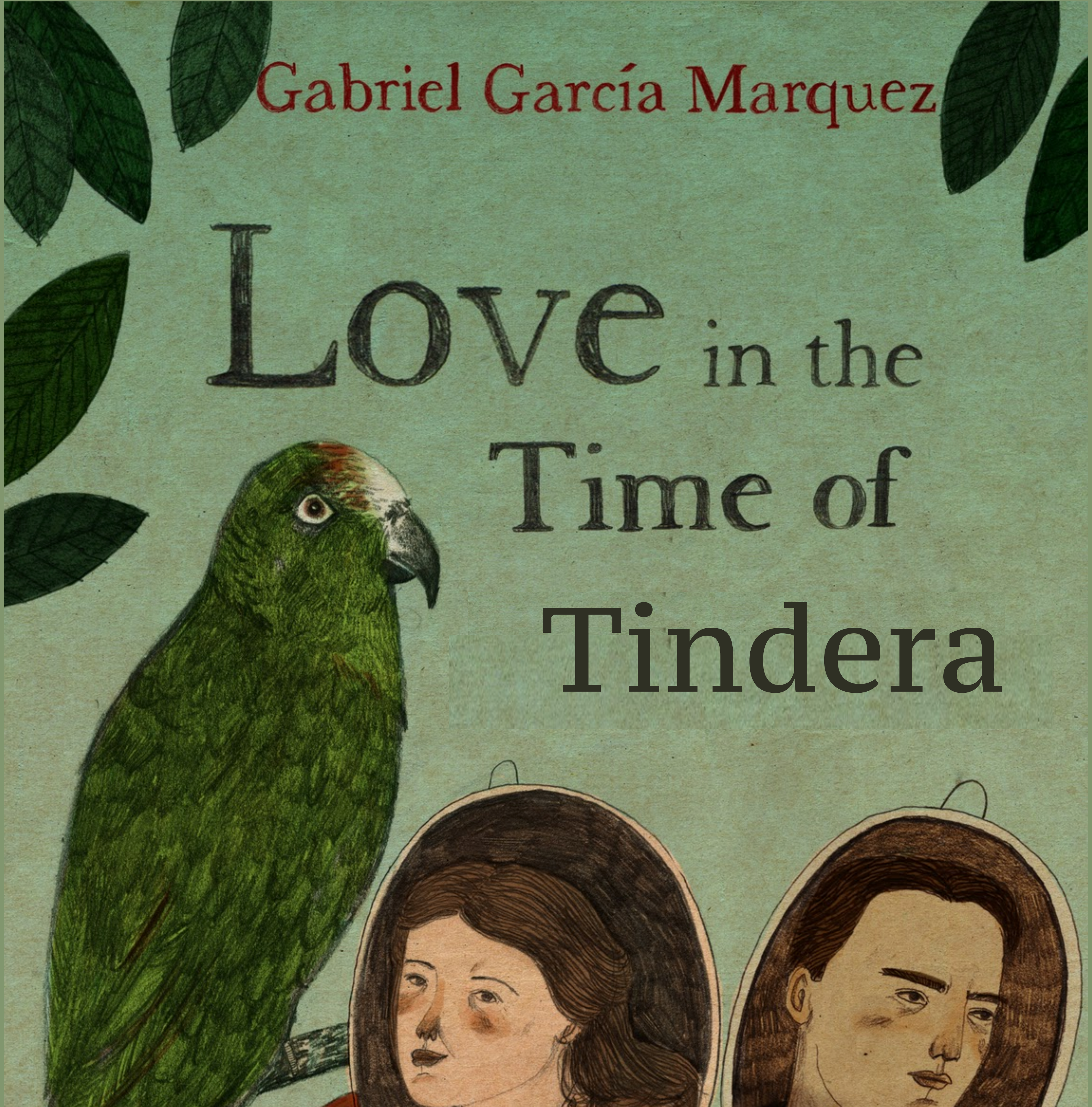
L18

4800

11.8.2016

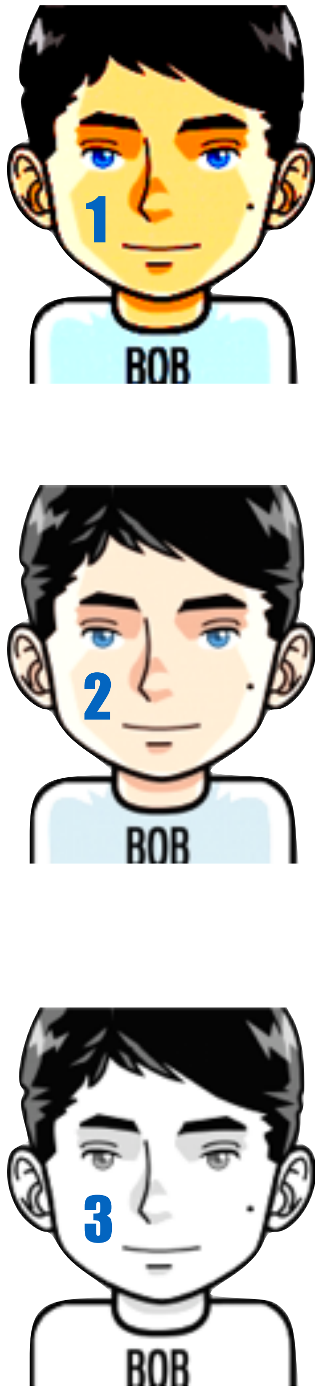
abhi shelat


Gabriel García Marquez
Love in the
Time of
Tindera

suitors

We have a group of suitors and reviewers

reviewers



2>1>3

2>3>1

1>3>2

Each has preferences over the other group

1>3>2

1>2>2

3>2>1

2>1>3

2>3>1

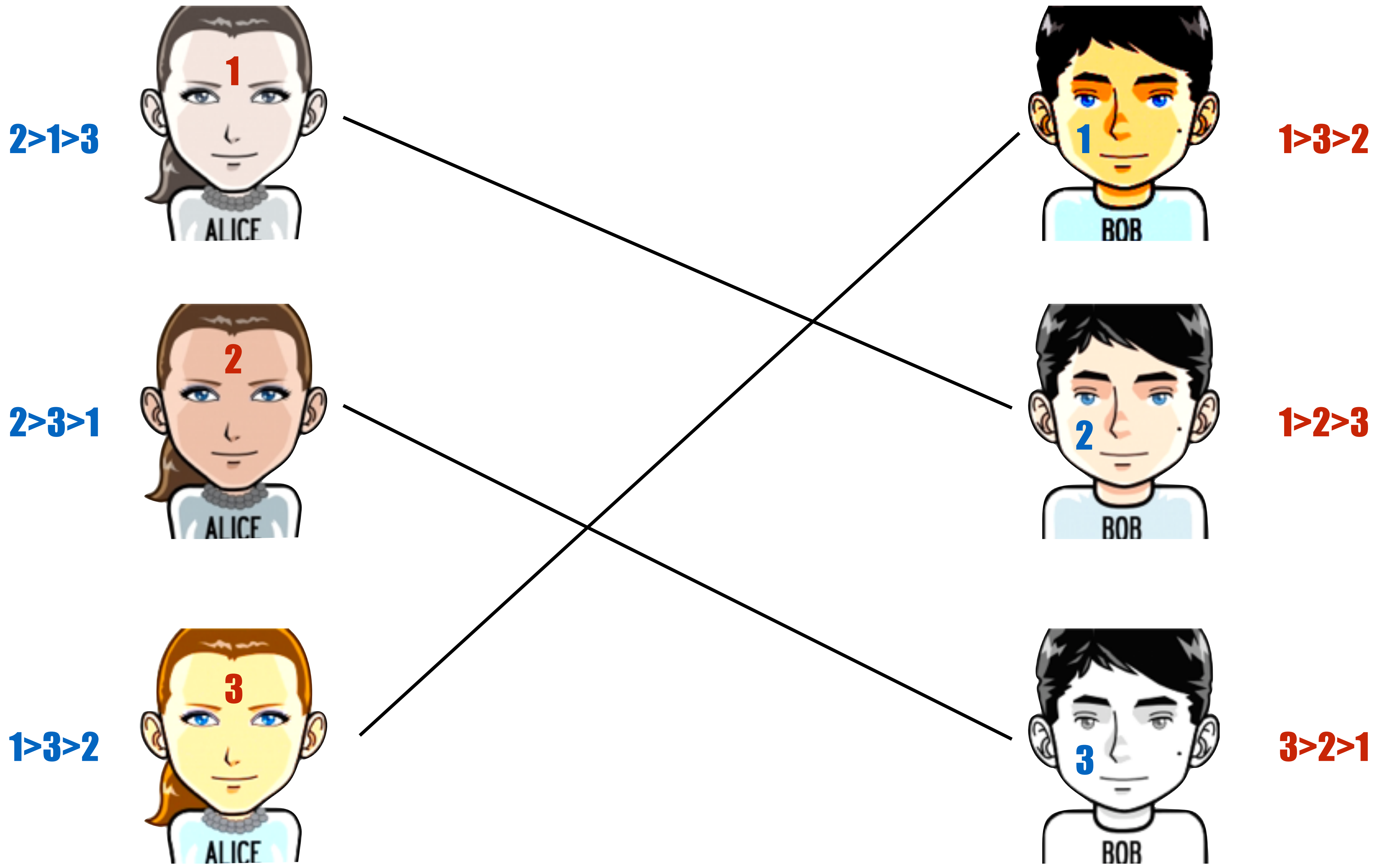
1>3>2

We seek a **stable matching** between the two

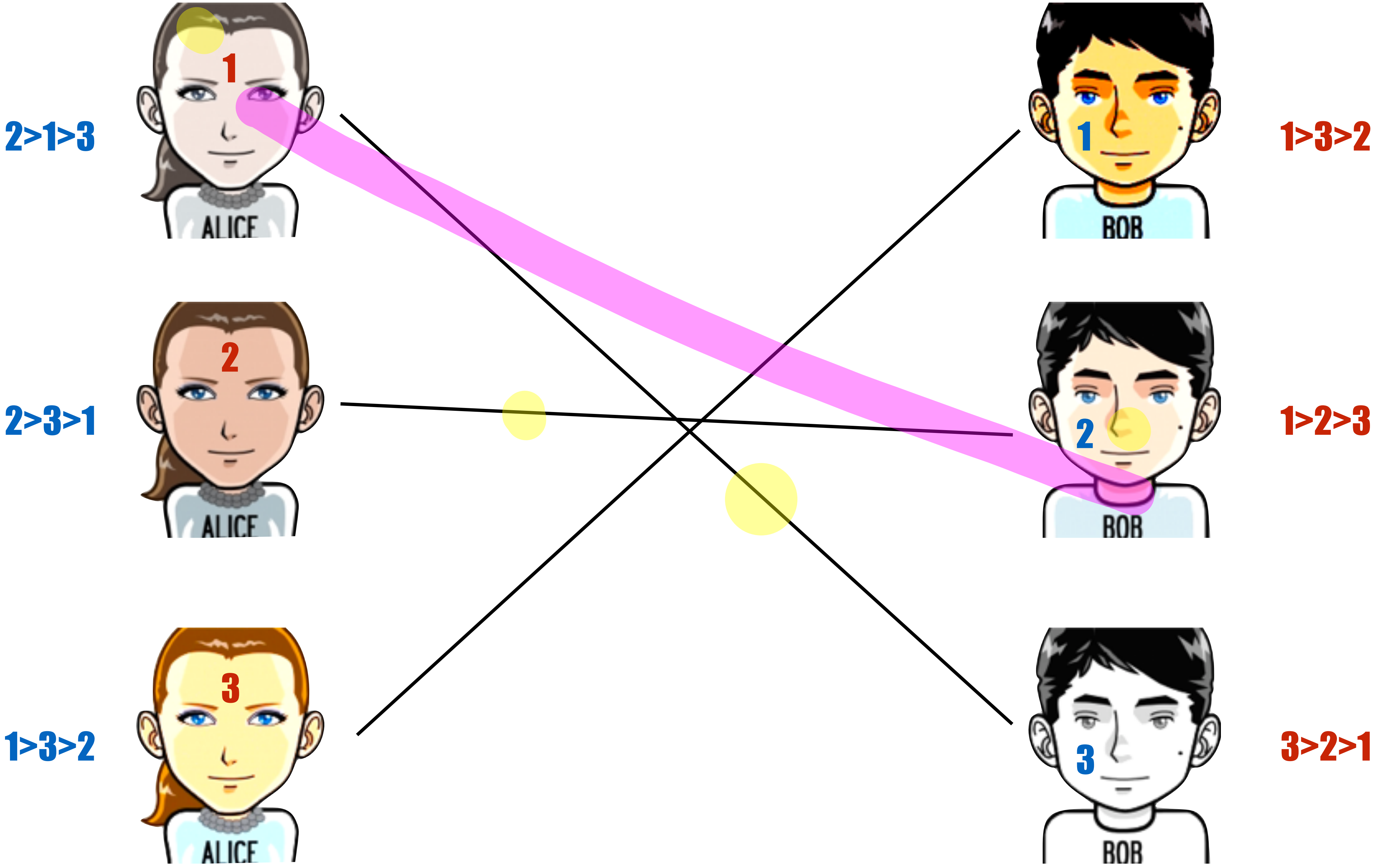
1>3>2

1>2>2

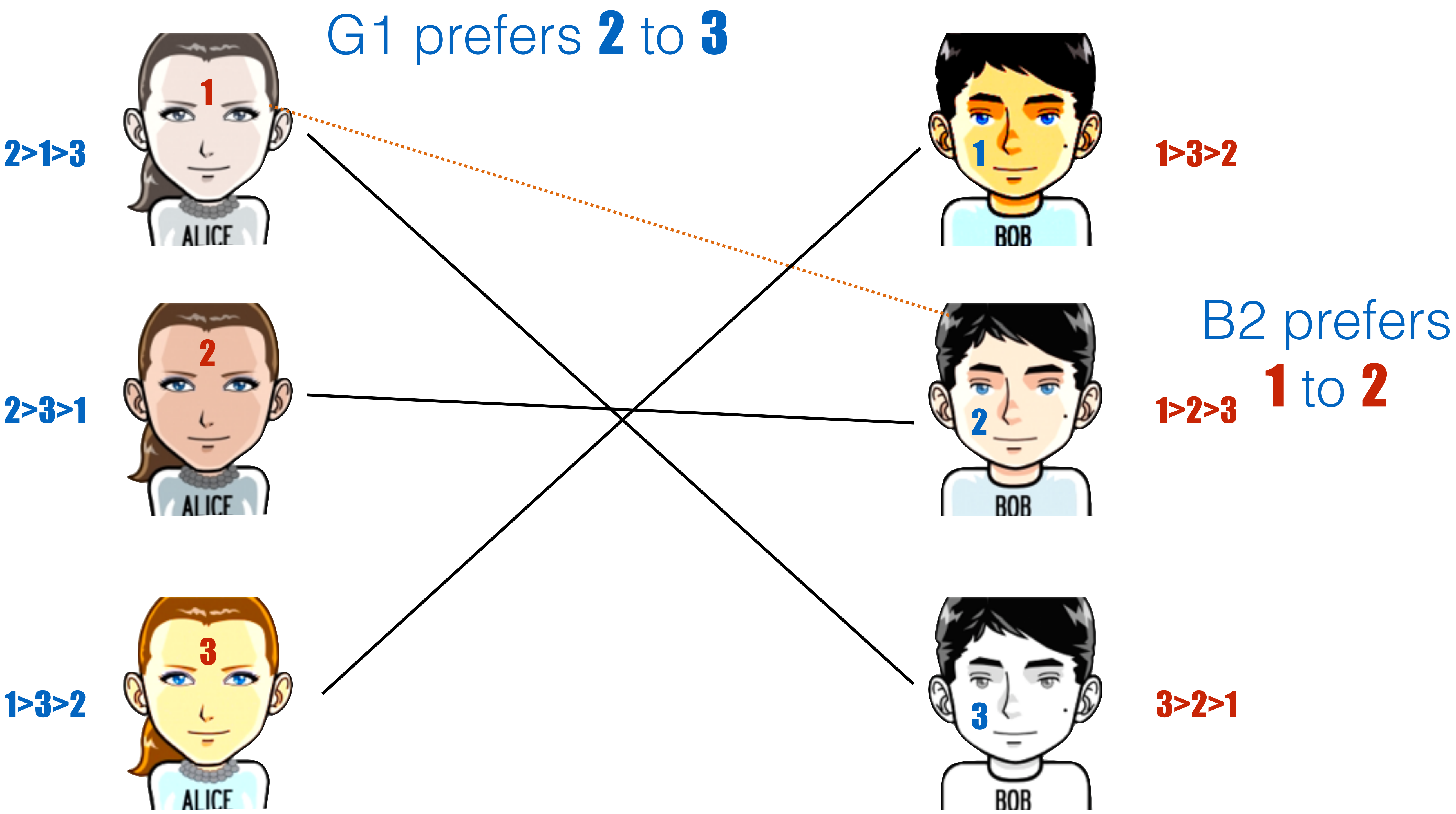
3>2>1



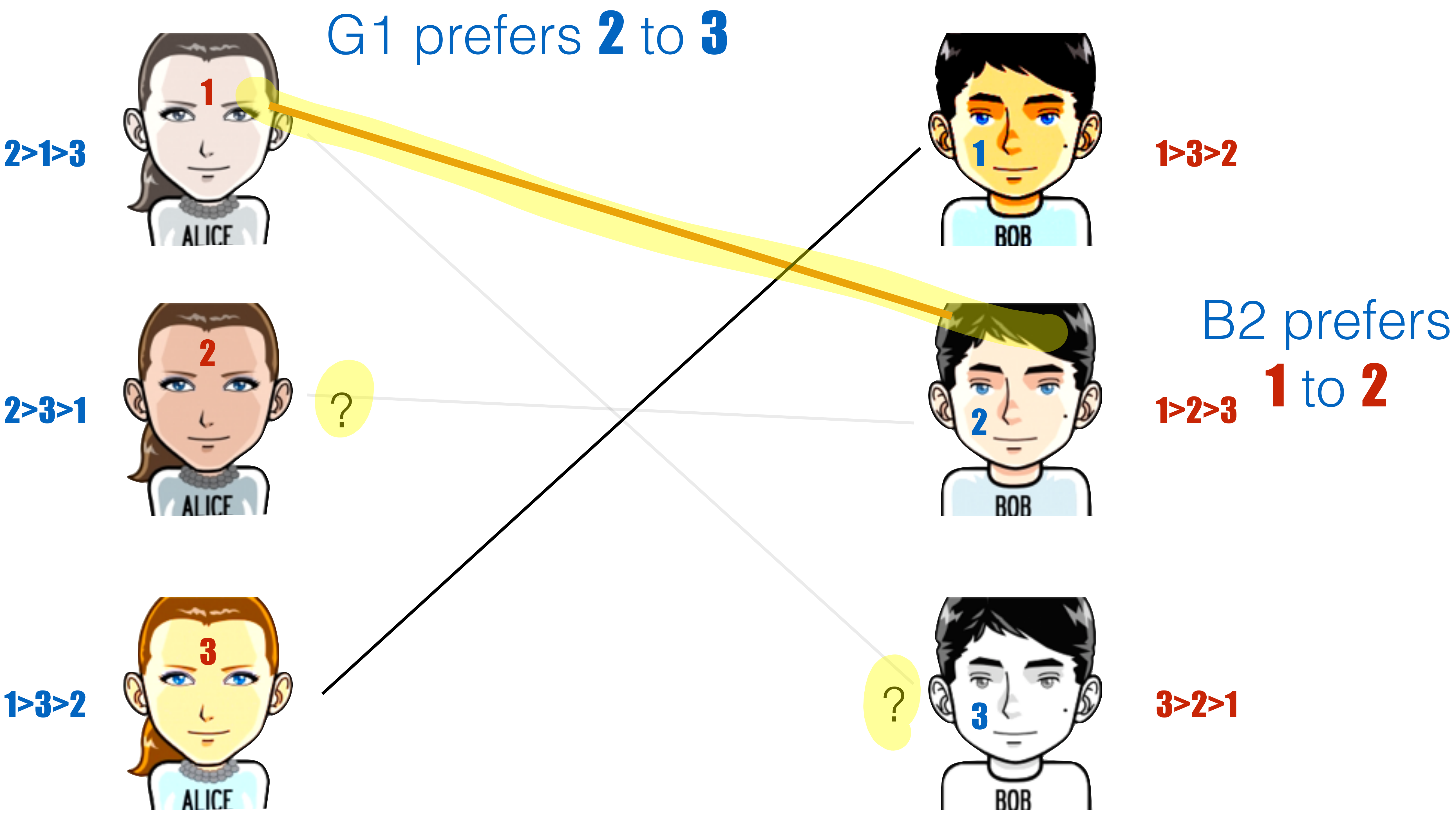
2>1>3
2>3>1
1>3>2
1
2
3
ALICE
ALICE
ALICE
1
2
3
BOB
BOB
BOB
1>3>2
1>2>3
3>2>1



Unstable Matching



Unstable Matching

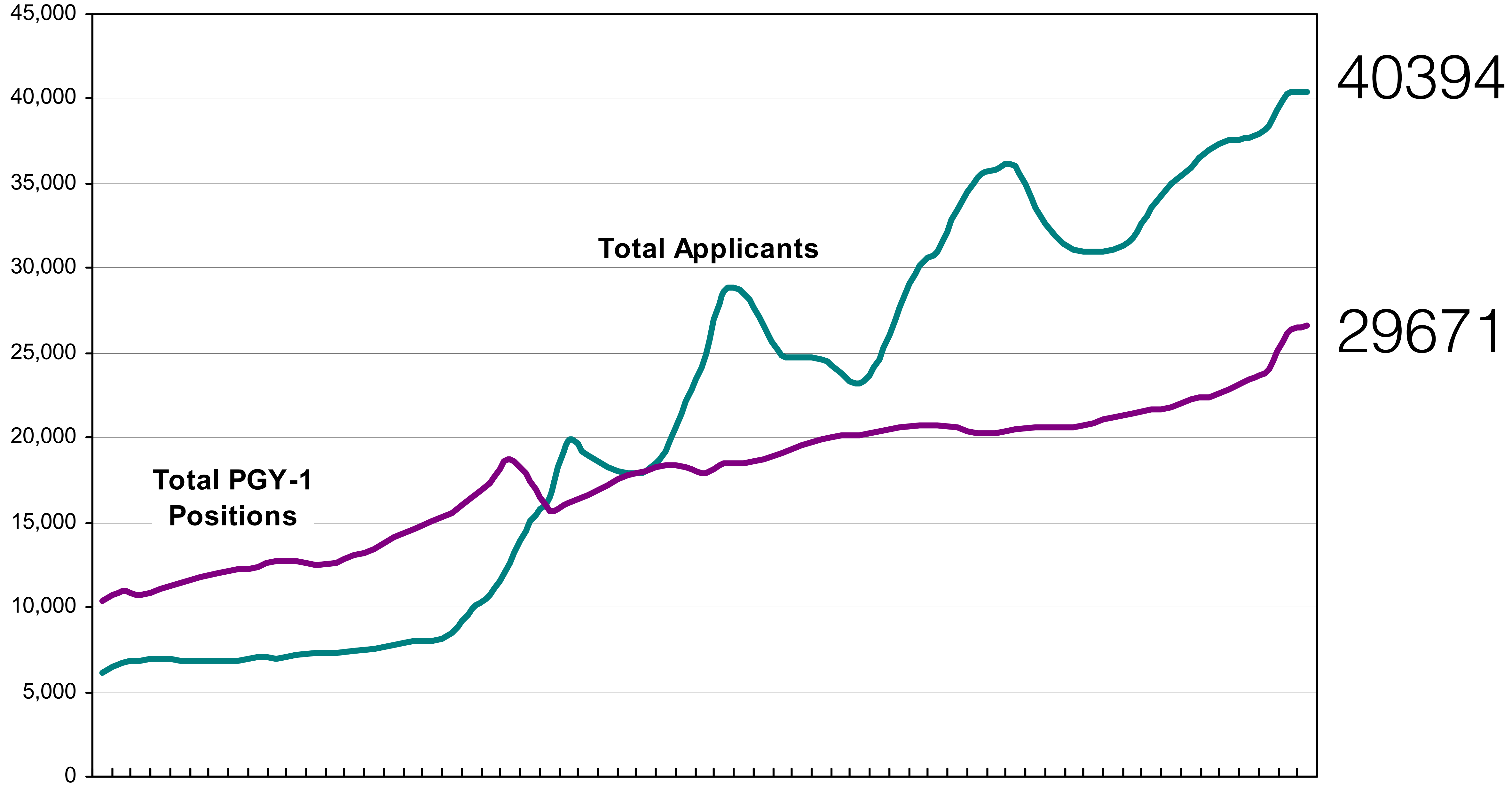


Unstable Matching

Stable Matching

Stable matching has many practical applications

Figure 1 Applicants and 1st Year Positions in The Match, 1952 - 2014



Applicant Type	Matched		
	2013 Graduates	Prior Year Graduates[1]	Total
CMG	2571	74	2645
IMG	146	353	499
USMG	23	2	25
TOTAL	**2740**	**429**	**3169**

UNIVERSITAS OSLOENSIS
MDCCCXI

NUS
National University of Singapore

SINGAPORE UNIVERSITY OF TECHNOLOGY AND DESIGN
Established in collaboration with MIT

NANYANG TECHNOLOGICAL UNIVERSITY


GreekYearbook
www.GreekYearbook.com
University of Virginia
Chi Omega Bid Day 2012

Definition: matchings

M= $\{m_1, m_2 \dots m_n\}$

W= $\{s_1 \dots s_n\}$

S= $\{(m_i, s_j)\}$ such that each m and each w appear in exactly one pair in S.

Definition: matchings

$$M = \{m_1, \ldots, m_n\}$$
$$W = \{w_1, \ldots, w_n\}$$

$$S = \{(m_{i_1}, w_{j_1}), \ldots, (m_{i_k}, w_{i_k})\}$$

Each m_i (w_i) appears only one in a pairing.
A matching is perfect if every m_i appears.

preferences

Image credits: Julia Nikolaeva

Definition: preferences

$$M = \{m_1, \ldots, m_n\}$$

$w_i <_{m_1} w_j$

"m_1 prefers w_j to w_i"

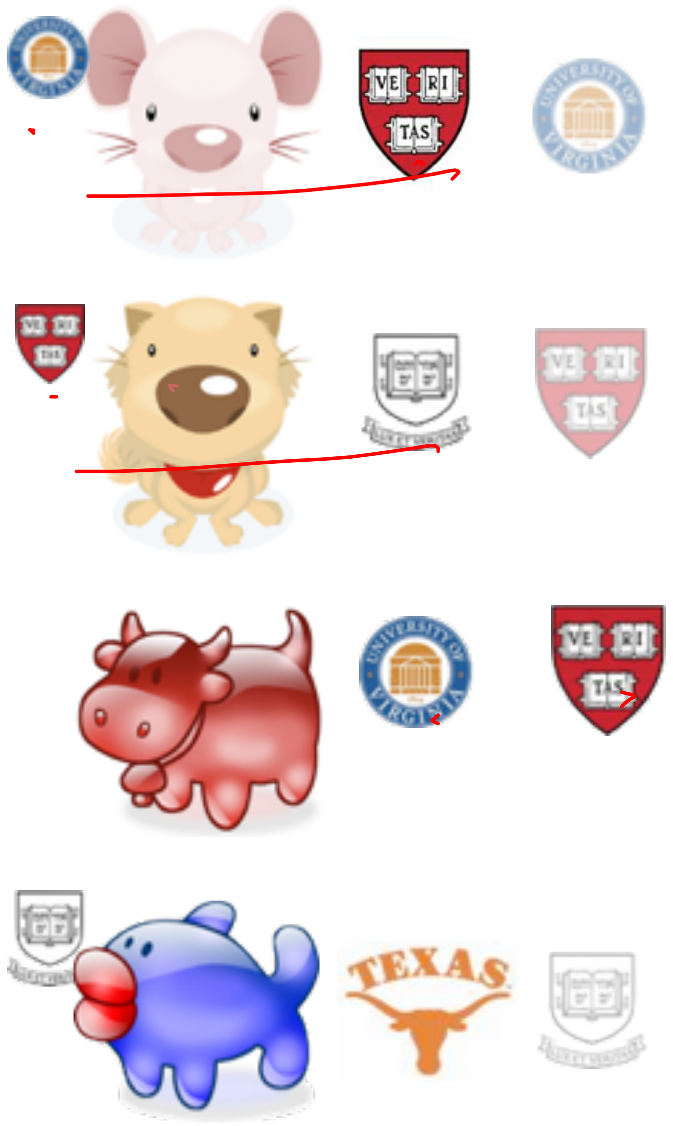


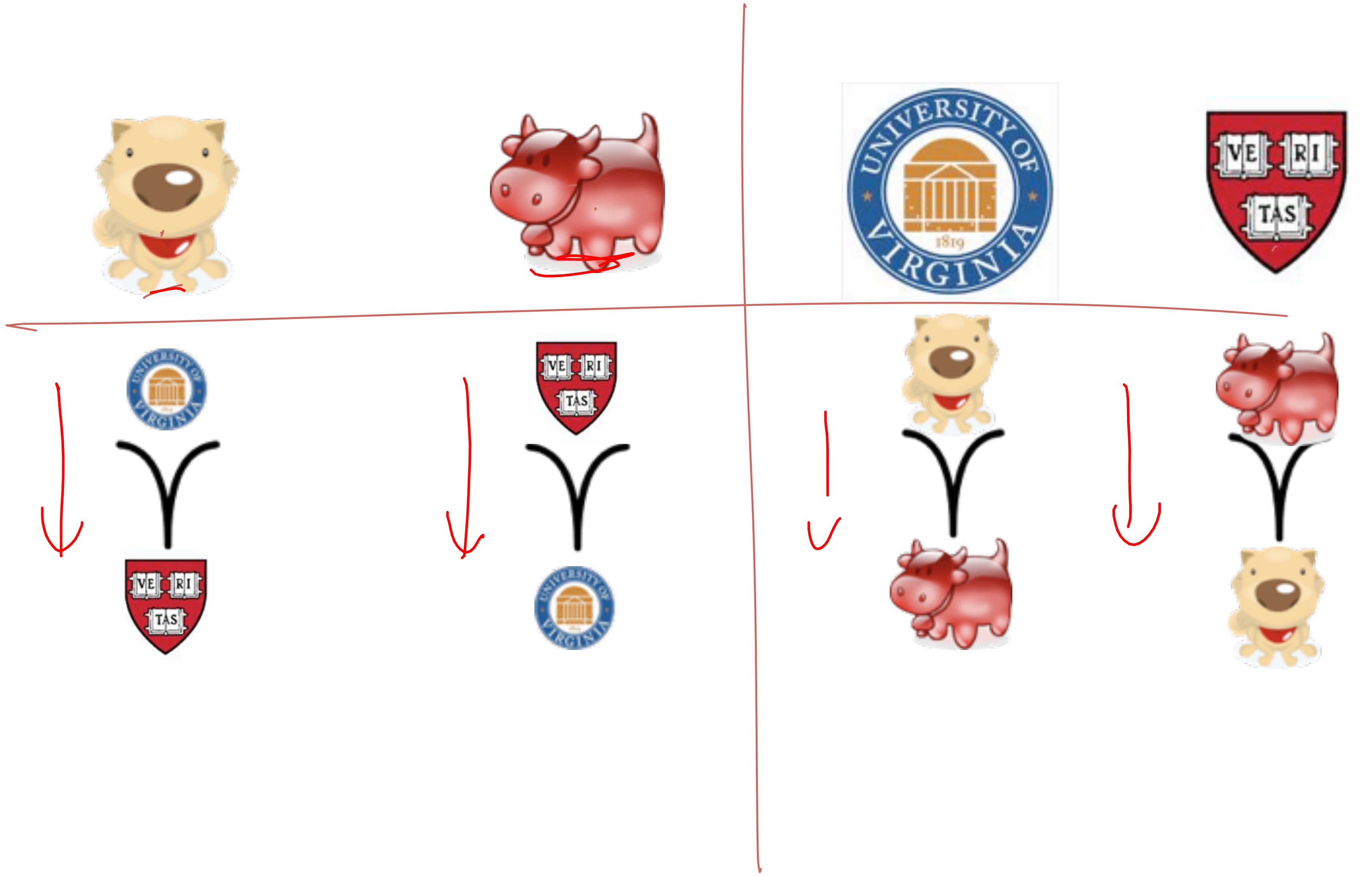
Image credits: Julia Nikolaeva

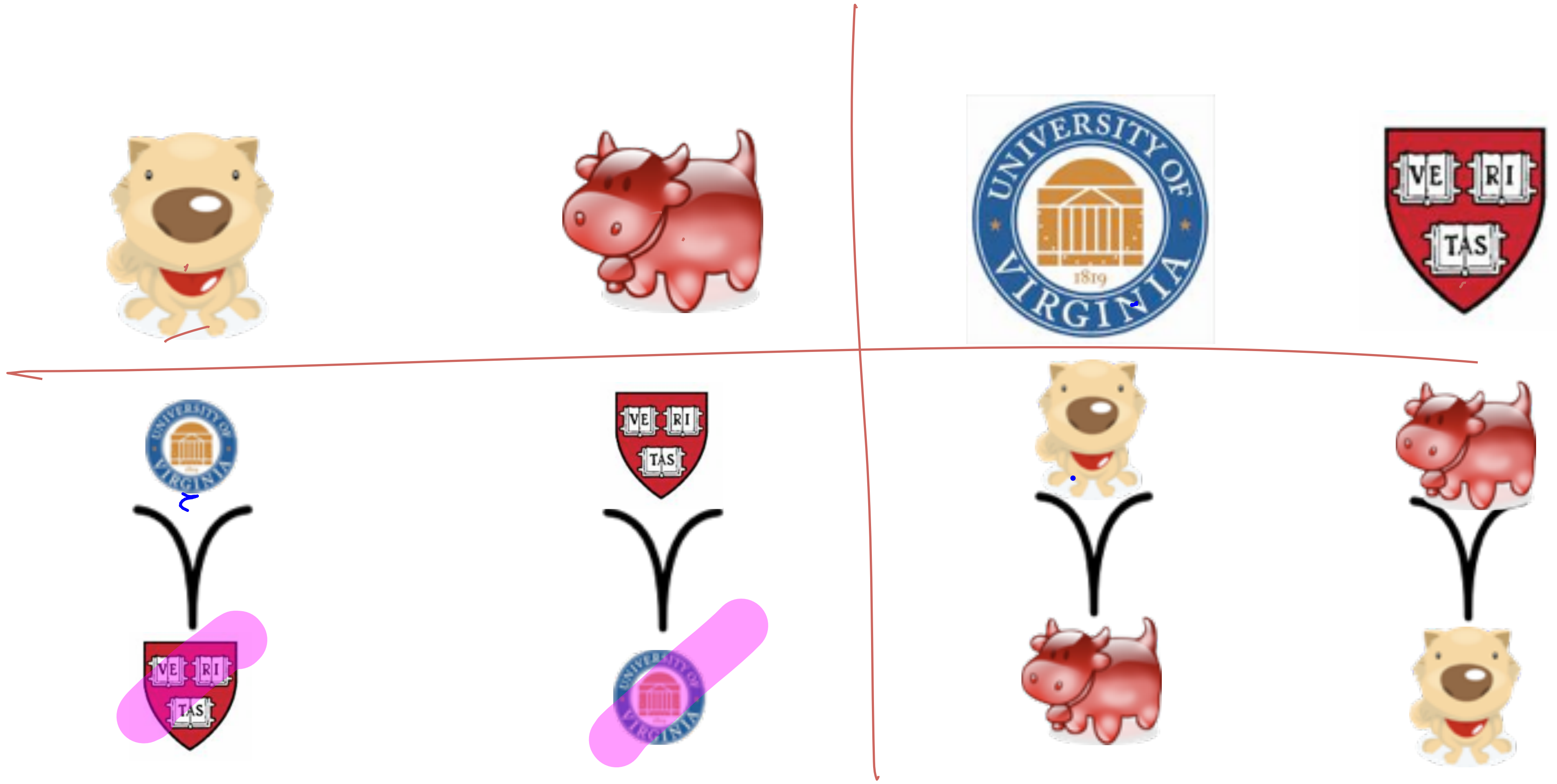
Example: preferences

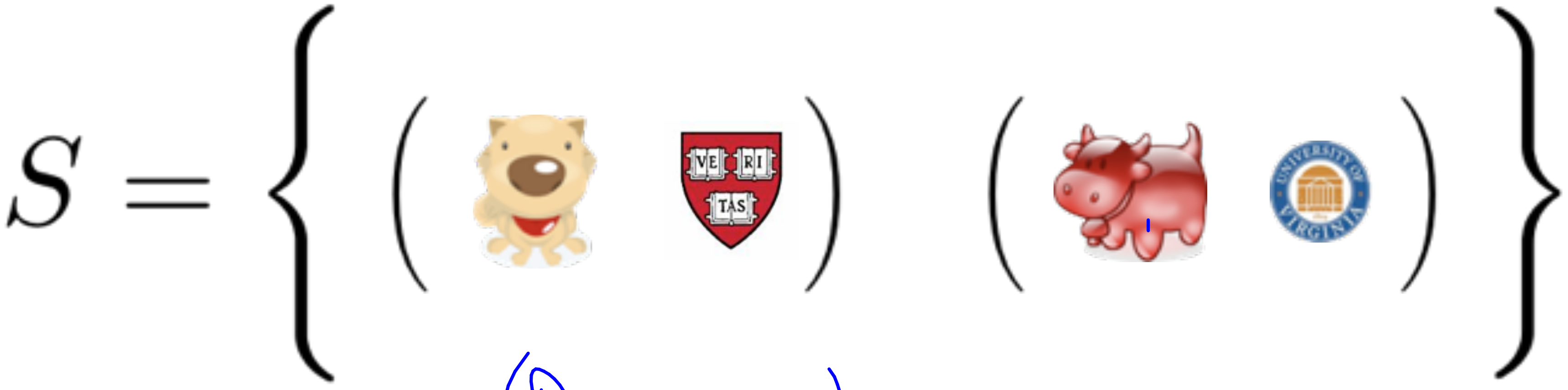
$$M = \{m_1, \ldots, m_n\}$$

m_i has a preference relation $\prec_{m_i}$ on the set W

$$w_1 \prec_{m_i} w_4 \prec_{m_i} w_2 \prec_{m_i} w_8 \cdots w_n$$

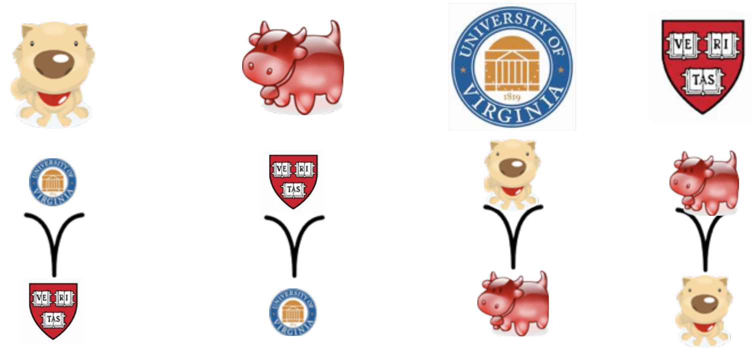

UNIVERSITY OF VIRGINIA
1819
VE RI TAS

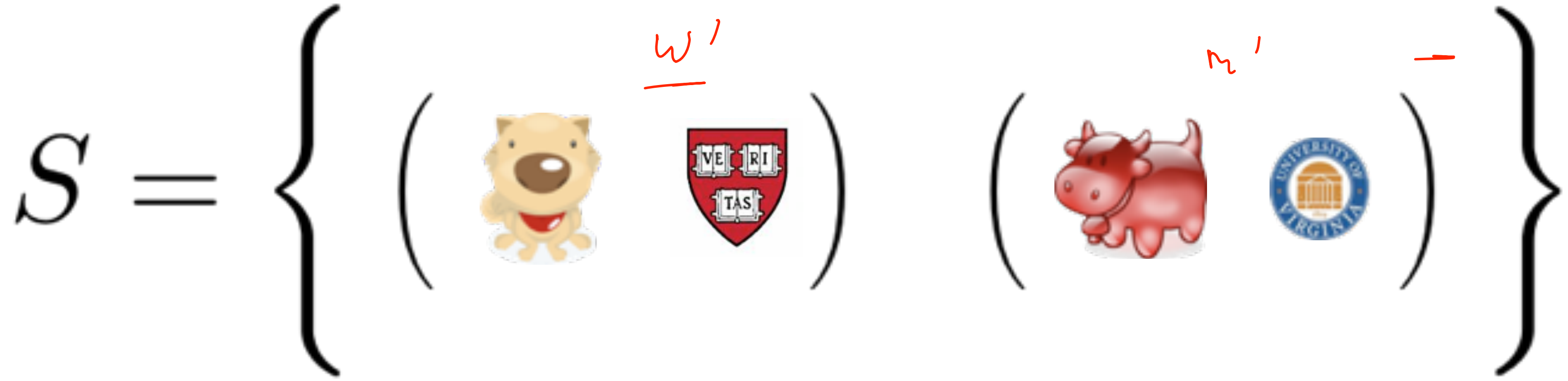




(Dog, UVA) is an instability

Def: instability





$w' <_m w$

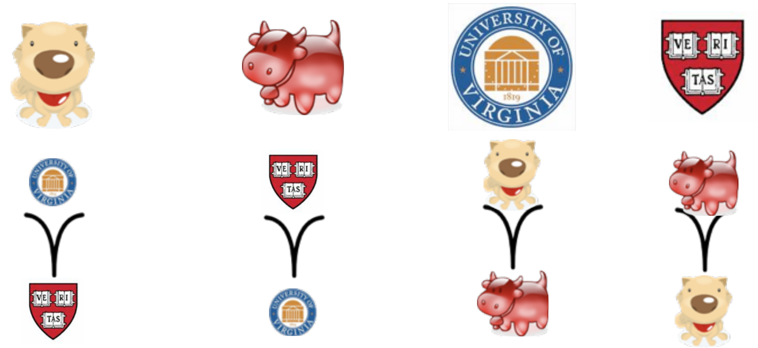
$m' <_w m$

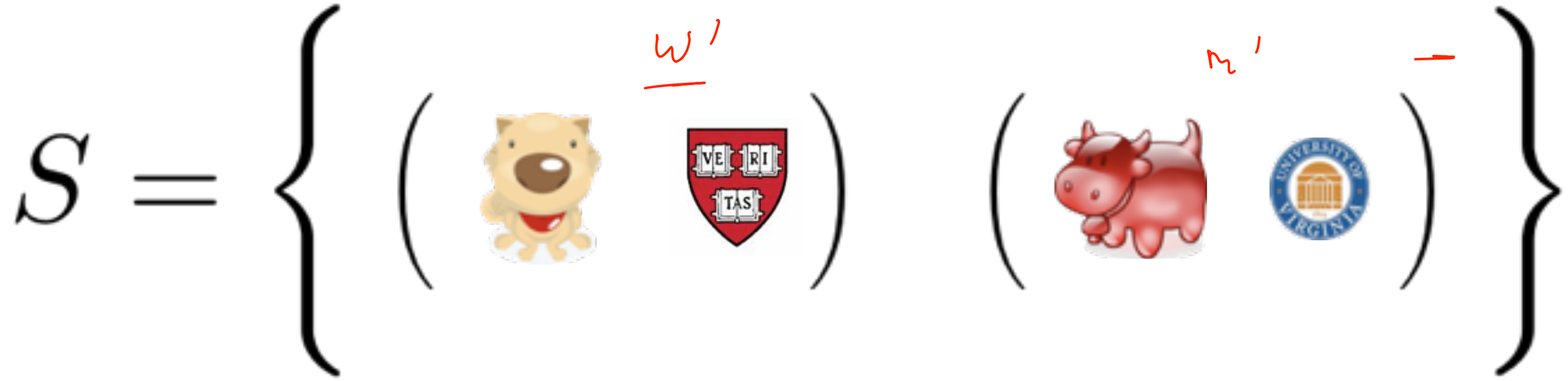
is an unmatched pair (m, w) such that

m prefers w to its current match w'

w prefers m to its current match m'

Def: instability





$$(m^*, w^*) \notin S$$

$$w' \prec_{m^*} w^*$$

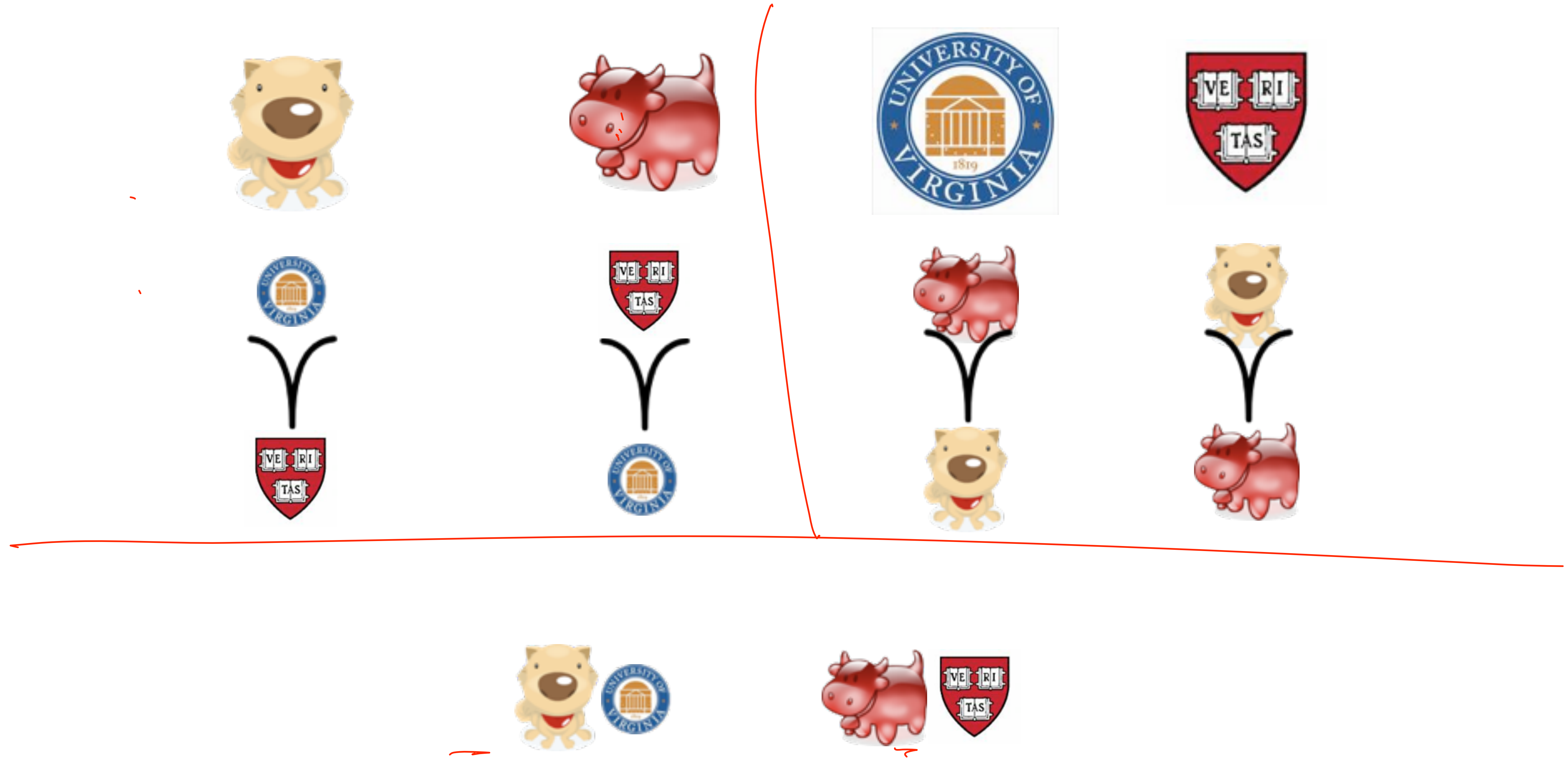
$$m' \prec_{w^*} m^*$$

M $= \{ (s_1,r_1), (s_2,r_2), \ldots (s_n,r_n) \}$

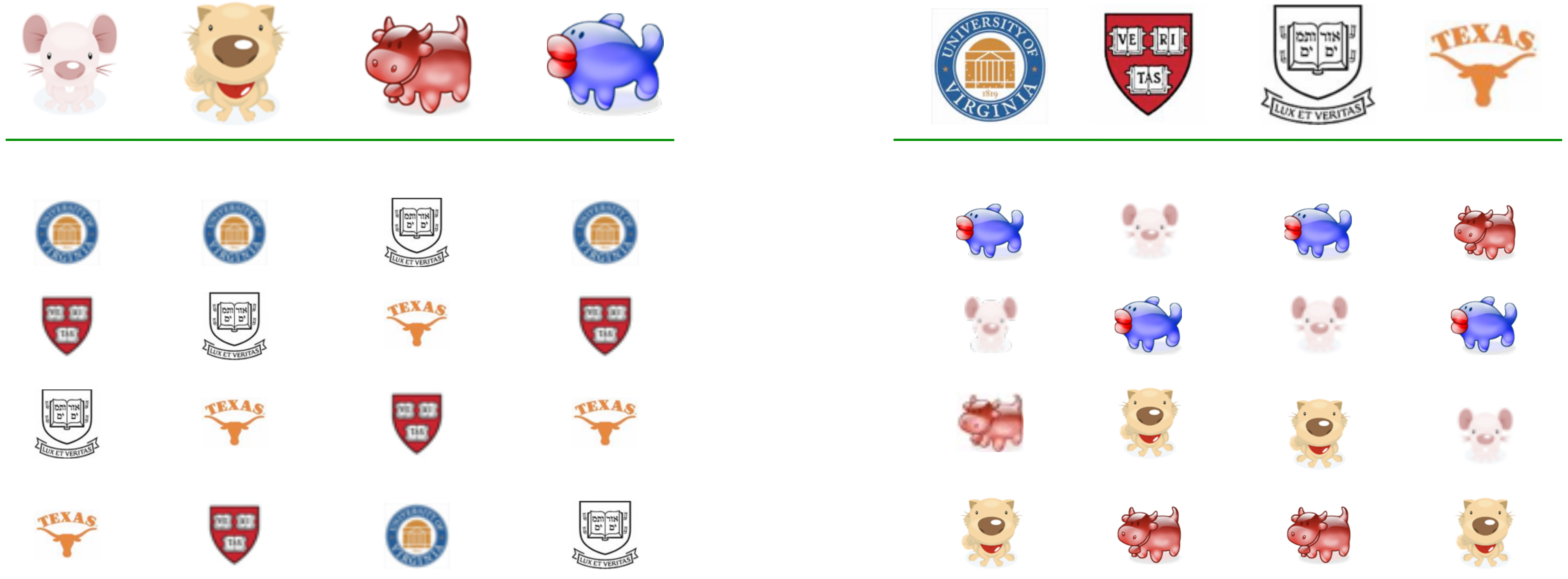
is a stable matching if

No unmatched pair (s*,r*) prefer each other to their partners in M

Example 2



Prove: for every input



there exists a stable matching.

proposal algorithm

- start with everyone unmatched

while there is an unmatched suitor S

Let r be highest ranked reviewer that S hasn't proposed to

S proposes a match with r

if r is unmatched or r is matched to (s', r) and s' $<_r$ s

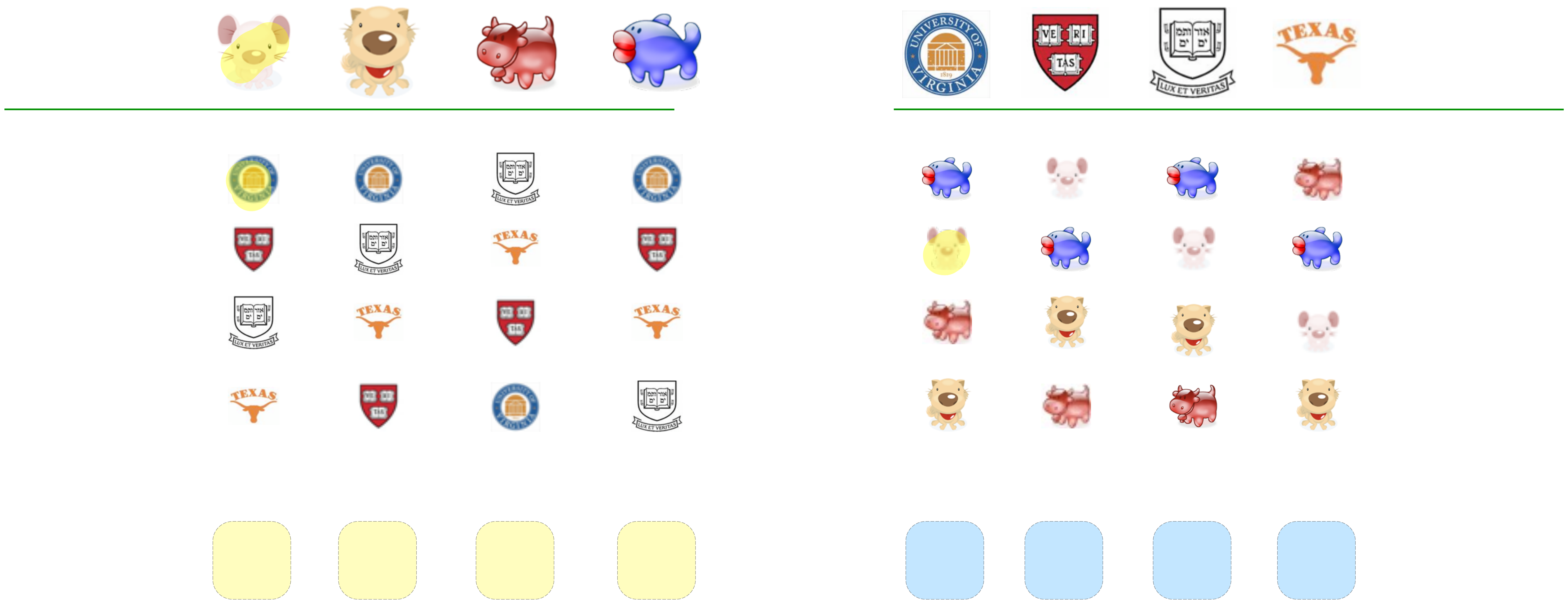
break the match (s', r) & create the match (s, r)

```
StableMatch(M, W, ≺_m, ≺_w)

Initialize all m, w to be FREE
while ∃FREE(m) and hasn't proposed to all W
      do Pick such an m
         Let w ∈ W be highest-ranked to whom m has not yet proposed
         if FREE(w)
            then Make a new pair (m, w)
         elseif (m′, w) is paired and m′ ≺_w m
               do Break pair (m′, w) and make m′ free
                  Make pair (m, w)
return Set of pairs
```

S

R

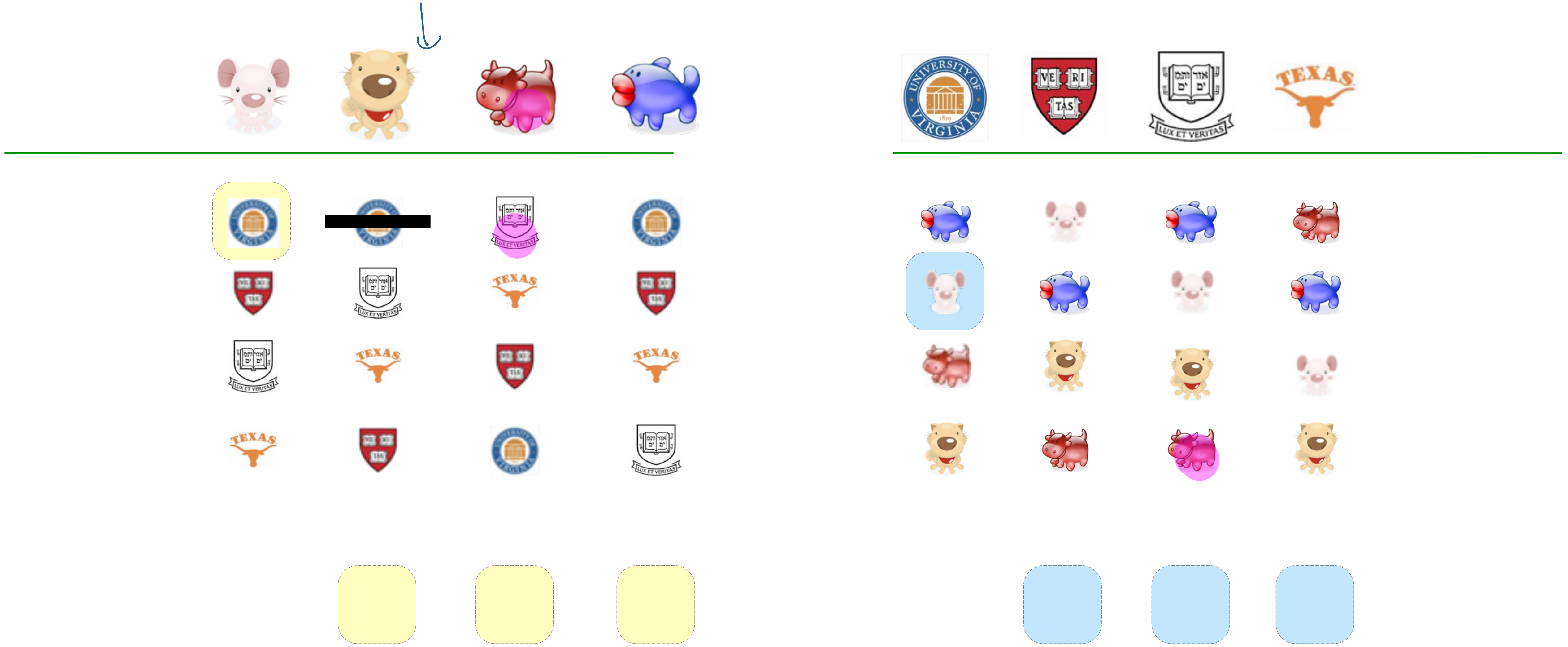


S

R

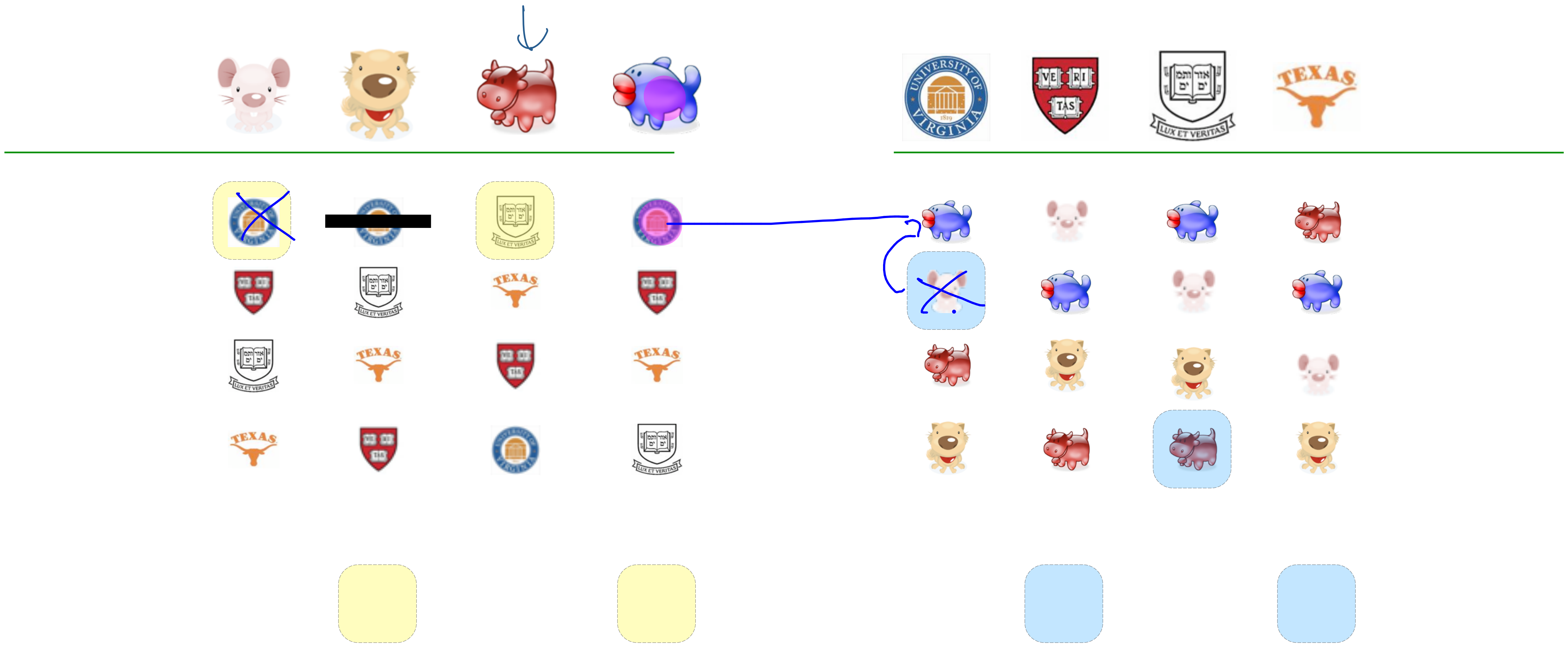
S

R

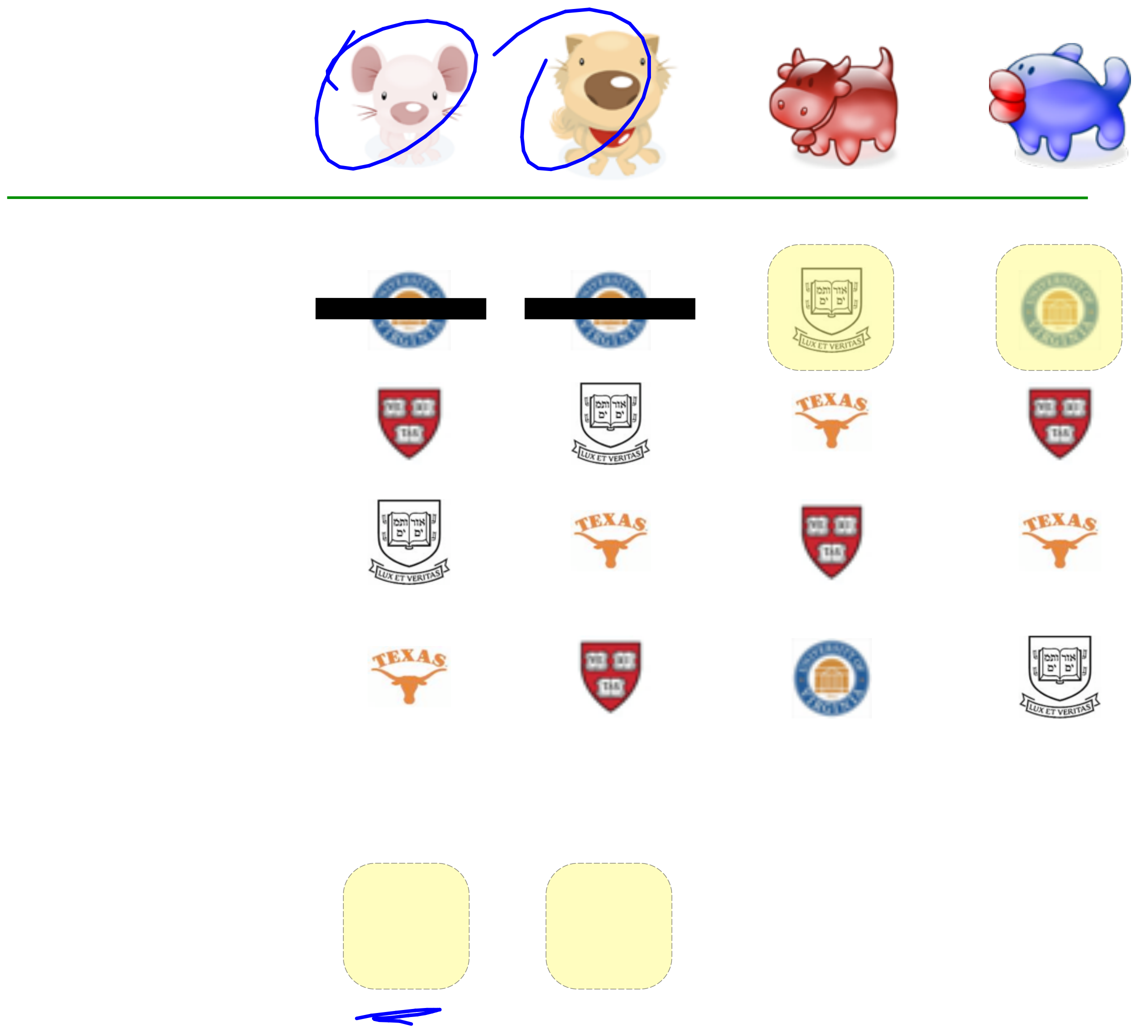


S

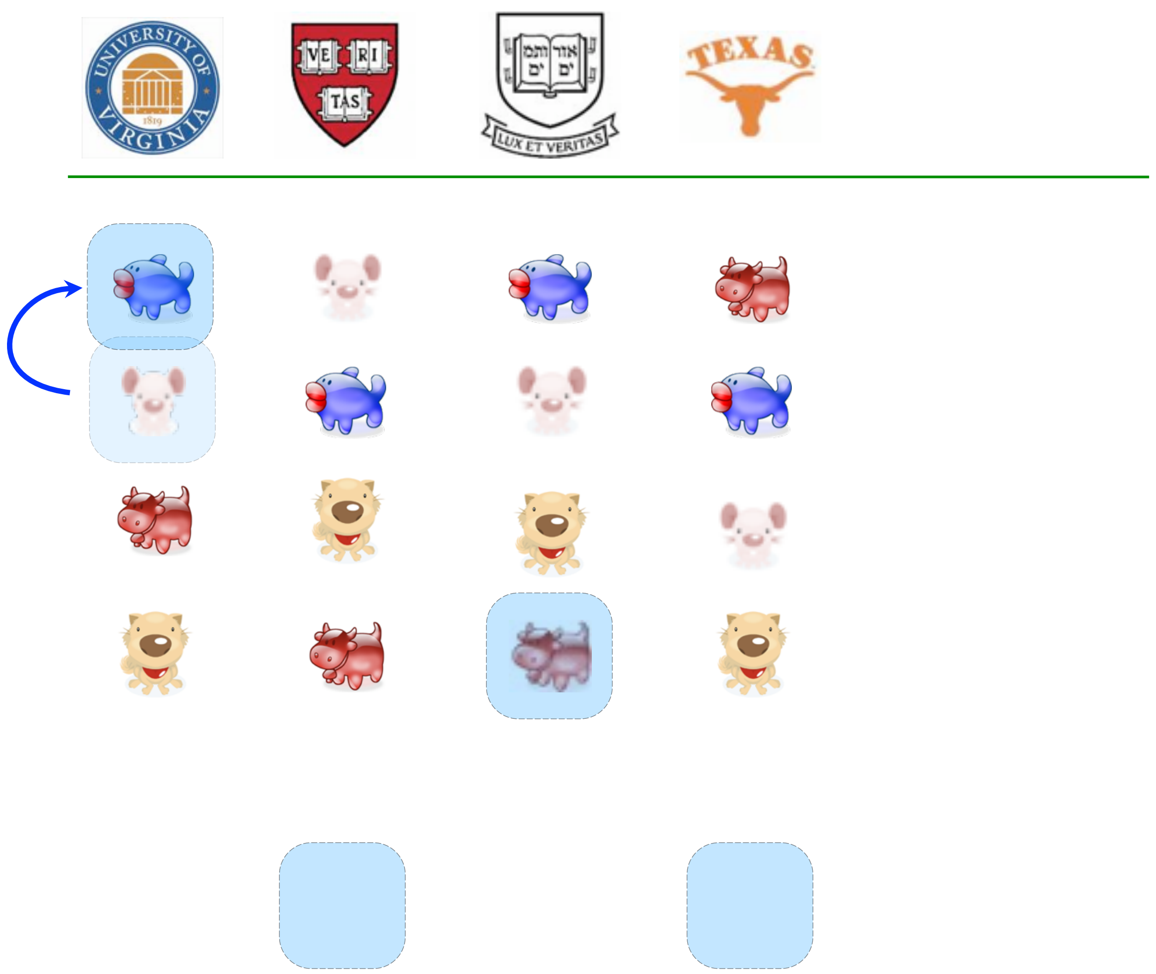
R



S

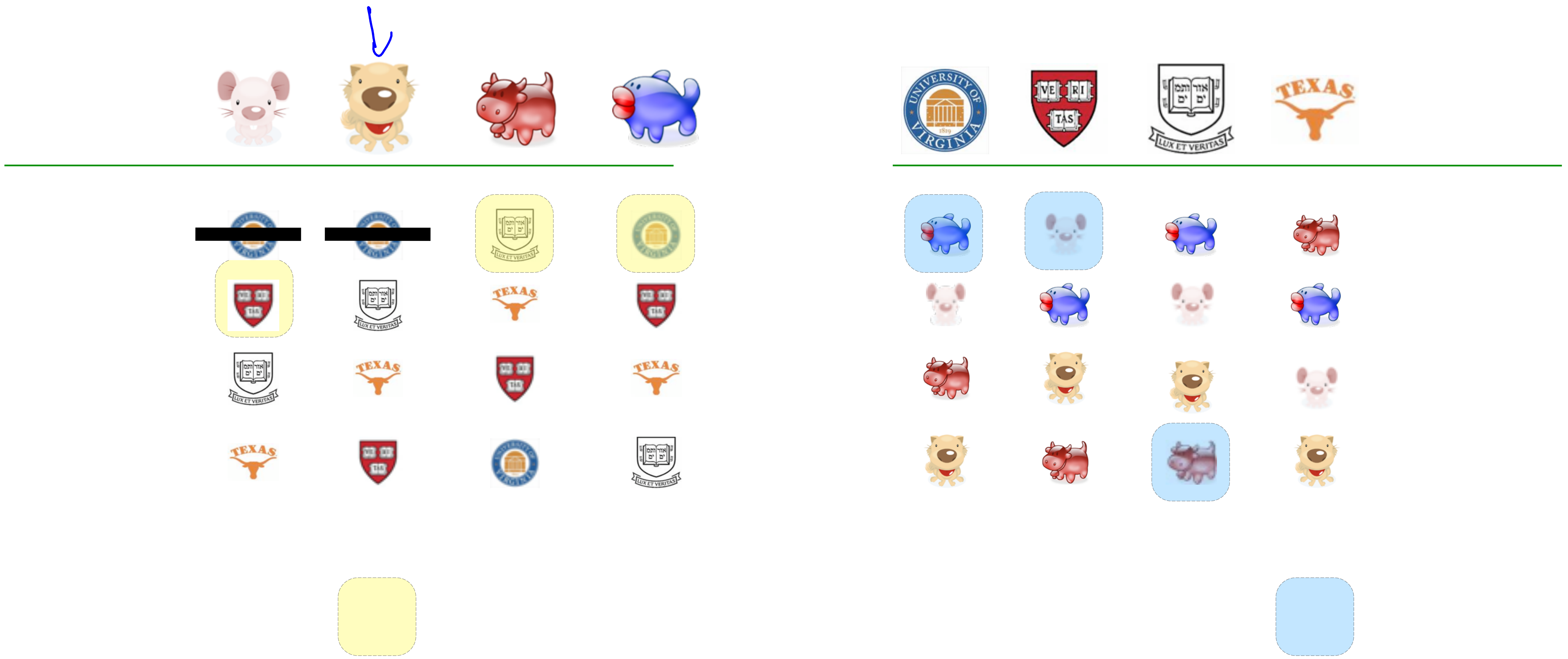


R



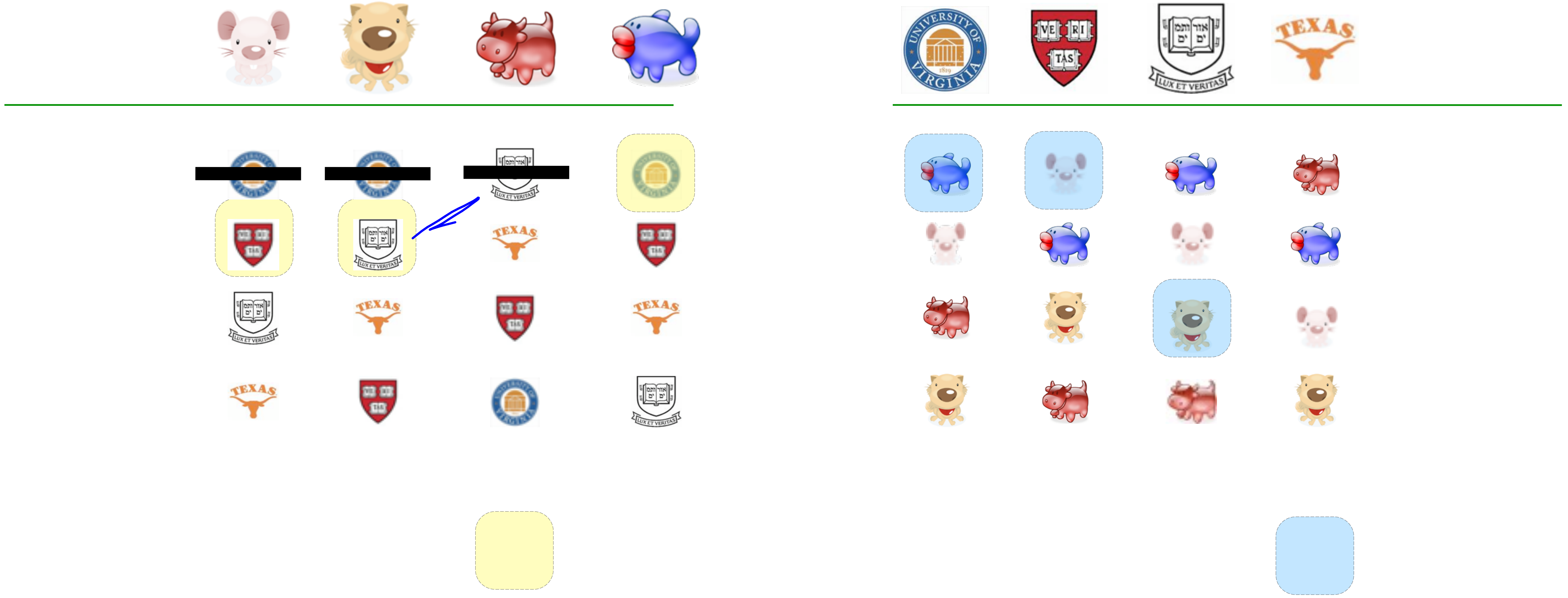
S

R

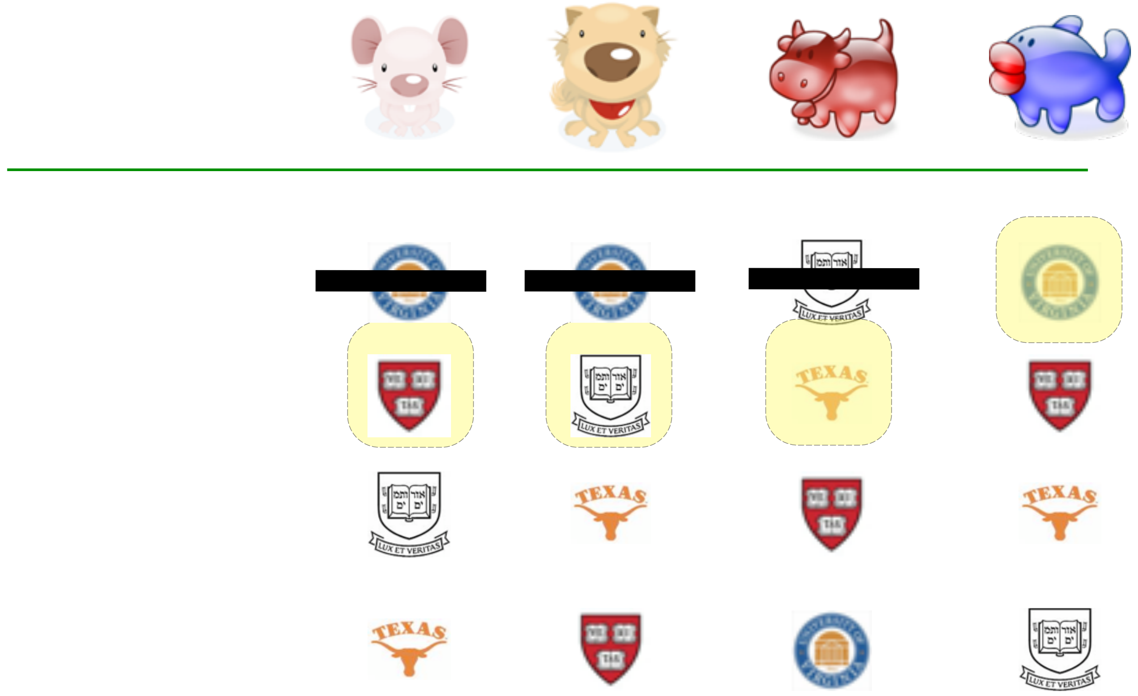


S

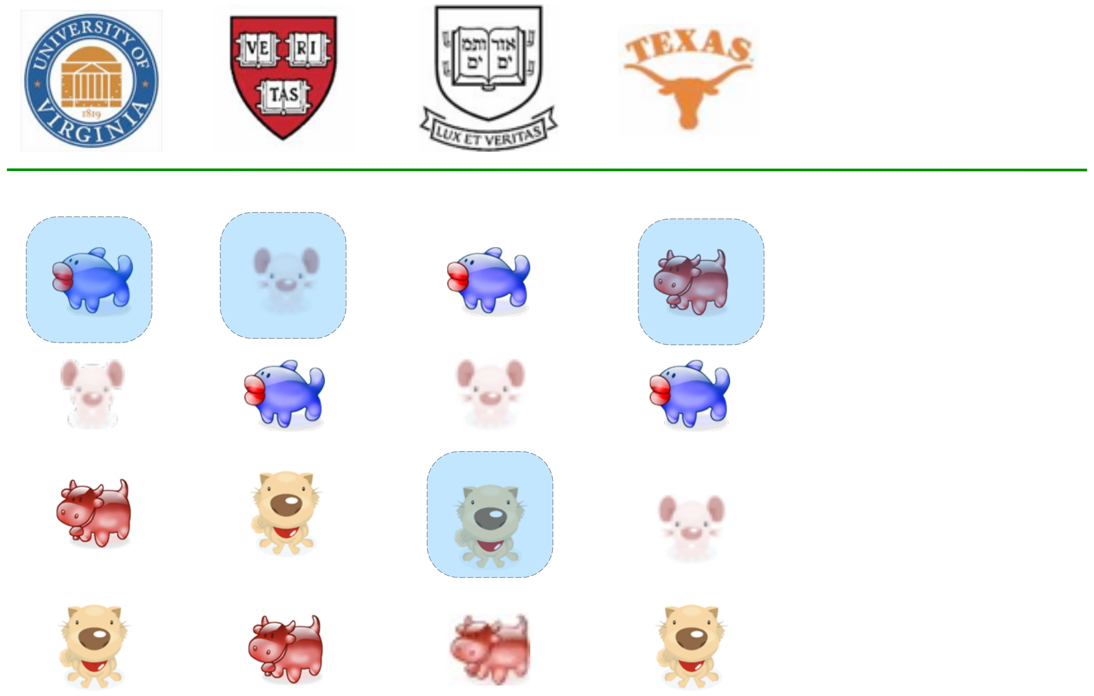
R



S



R



Proposal algorithm ends

Proposal algorithm ends

$O(n^2)$ steps

each m proposes at most once to each w.

each m proposes at most n times.

size of M is at most n.

output is a matching

Each m / w only appears at most once in the output.

By lines 6 and 9, when a match is added to potential output, both parties are unmatched at the time of match by lines 2, 5 and/or 8.

```
StableMatch(M, W, ≺_m, ≺_w)
Initialize all m, w to be FREE
while ∃FREE(m) and hasn't proposed to all W
      do Pick such an m
         Let w ∈ W be highest-ranked to whom m has not yet proposed
         if FREE(w)
            then Make a new pair (m, w)
         elseif (m′, w) is paired and m′ ≺_w m
               do Break pair (m′, w) and make m′ free
                  Make pair (m, w)
return Set of pairs
```

StableMatch$(M, W, \prec_m, \prec_w)$

Initialize all m, w to be free
while $\exists$free(m) and hasn't proposed to all W
    do Pick such an m
        Let $w \in W$ be highest-ranked to whom m has not yet proposed
        if free(w)
            then Make a new pair (m, w)
        elseif (m', w) is paired and $m' \prec_w m$
            do Break pair (m', w) and make m' free
                Make pair (m, w)
return Set of pairs

output is perfect

$|M| = n.$ Because

$\Rightarrow$ if there is an unmatched suitor

$\Rightarrow$ $\exists$ an unmatched reviewer.

(so alg has not terminated yet)

output is perfect

if $\exists m$ who is free, then

$\exists w$ who has not been asked

output is stable

Proof: By contradiction. Spse output is not stable. There exists an unmatched pair (m^*, w^*) such that $w <_{m^*} w^*$ and $m <_{w^*} m^*$.

and (m^*, w) $(m, w^*) \in M$

output is stable

spse not. $\exists (m^*, w), (m, w^*) \in S$ $\quad w \prec_{m^*} w^*$ $\quad m \prec_{w^*} m^*$

Consider the moment when w^* is matched with m, and the moment when m^* is matched with w.

① m^* must have proposed to w last. But we know that m^* preferred w^* to w. And by the algorithm, this means that m^* proposal to w^* before proposing to w.

② What happened when m^* proposed to w^* ??

ⓐ (m^*, w^*) was made but then at some point (m, w^*) was made

or ⓑ w^* was already matched to m' and $m^* <_{w^*} m'$

In both cases, this suggests $m^* <_{w^*} m$ which contradicts [highlighted] above.

output is stable

spse not. $\exists (m^*, w), (m, w^*) \in S$ $w \prec_{m^*} w^*$ $m \prec_{w^*} m^*$

m* last proposal was to w

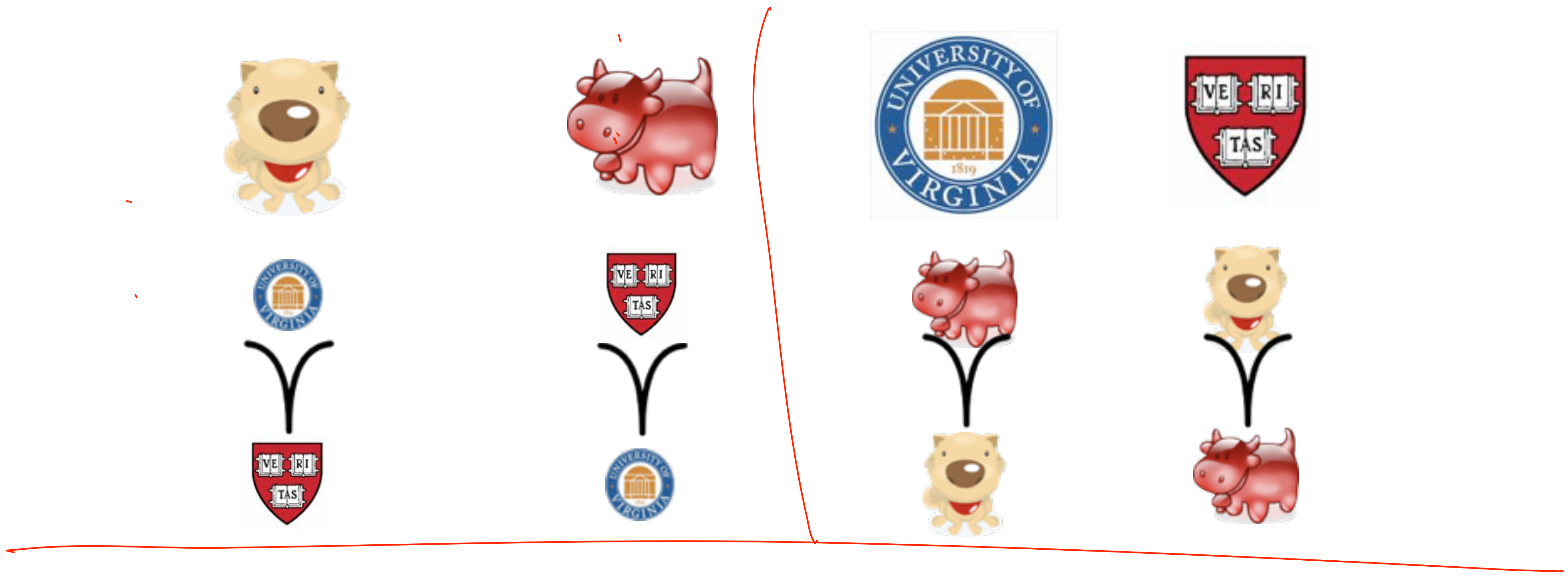
but $w \prec_{m^*} w^*$ and so m* must have already asked w*

and must have been rejected by $m^* \prec_{w^*} m'$

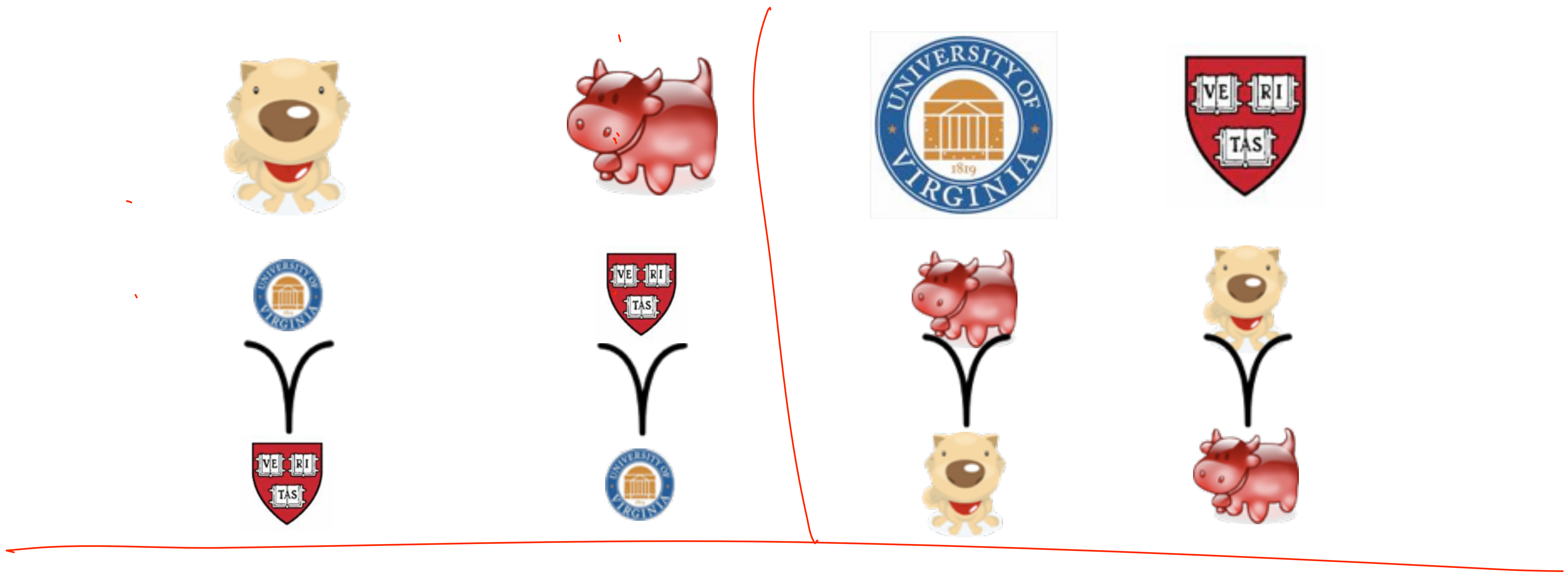
then either $m' \prec_{w^*} m$ or m'=m

which contradicts assumption $m \prec_{w^*} m^*$

Proposer wins



Proposer wins



Remarkable theorem

w is valid for m: if $\exists$ a stable matching S such that $(m,w) \in S$.

best(m): best(m) is valid for m and there is no valid w^* such that best(m) $<_m w^*$

Thm: G-S returns the match $\{(m, \text{best}(m))\}$.

(proposer optimal match)

suitor

GS is Suitor-optimal.

Proof: Suppose that GS did not return the $S^* = \{(m, best(m))\}$.

It returned $S \neq S^*$. i.e, there is some m, (a) $w = best(m)$.

suitor optimal → S^* $\qquad$ S

$(M, \underline{w})$ $\qquad$ (a) $w' <_m w$ b/c $w = best(m)$ $\qquad$ $(M, \underline{w'})$

$(m', \underline{w'})$ $\qquad$ (b) $m' <_w m$ $\qquad$ (m', w)

why??

since (M, w) was a valid match,

w must prefer m

Conclusion: S was not stable b/c of (m, w).

⇒ contradiction. to the underlined sentence.

GS matching vs R-opt

S1	S2	S3	S4

R1	R2	R3	R4
S1	S1	S1	S1
S2	S2	S2	S2
S3	S3	S3	S3
S4	S4	S4	S4

S1	S2	S3	S4
R1	R1	R1	R1
R2	R2	R2	R2
R3	R3	R3	R3
R4	R4	R4	R4

R1	R2	R3	R4
S1	S1	S1	S1
S2	S2	S2	S2
S3	S3	S3	S3
S4	S4	S4	S4

S1	S2	S3	S4
R1	R1	R1	R1
R2	R2	R2	R2
R3	R3	R3	R3
R4	R4	R4	R4

R1	R2	R3	R4
S1	S1	S1	S1
S2	S2	S2	S2
S3	S3	S3	S3
S4	S4	S4	S4

Not honest

S1	S2	S3
R2	R1	R1
R1	R2	R3
R3	R3	R2

R1	R2	R3
S1	S2	S2
S2	S1	S3
S3	S3	S1

Not honest

S1	S2	S3
R2	R1	R1
R1	R2	R3
R3	R3	R2

R1	R2	R3
S1	S2	S2
S2	S1	S3
S3	S3	S1

~~R2~~	~~R1~~	~~R1~~
R1	R2	R3
R3	R3	R2

S1	S2	S2
S3	S1	S3
S2	S3	S1

Not honest

S1	S2	S3
R2	R1	R1
R1	R2	R3
R3	R3	R2

R1	R2	R3
S1	S2	S2
S2	S1	S3
S3	S3	S1

R2	R1	R1
R1	R2	R3
R3	R3	R2

S1	S2	S2
S3	S1	S3
S2	S3	S1

THE MATCH℠

NATIONAL RESIDENT MATCHING PROGRAM®

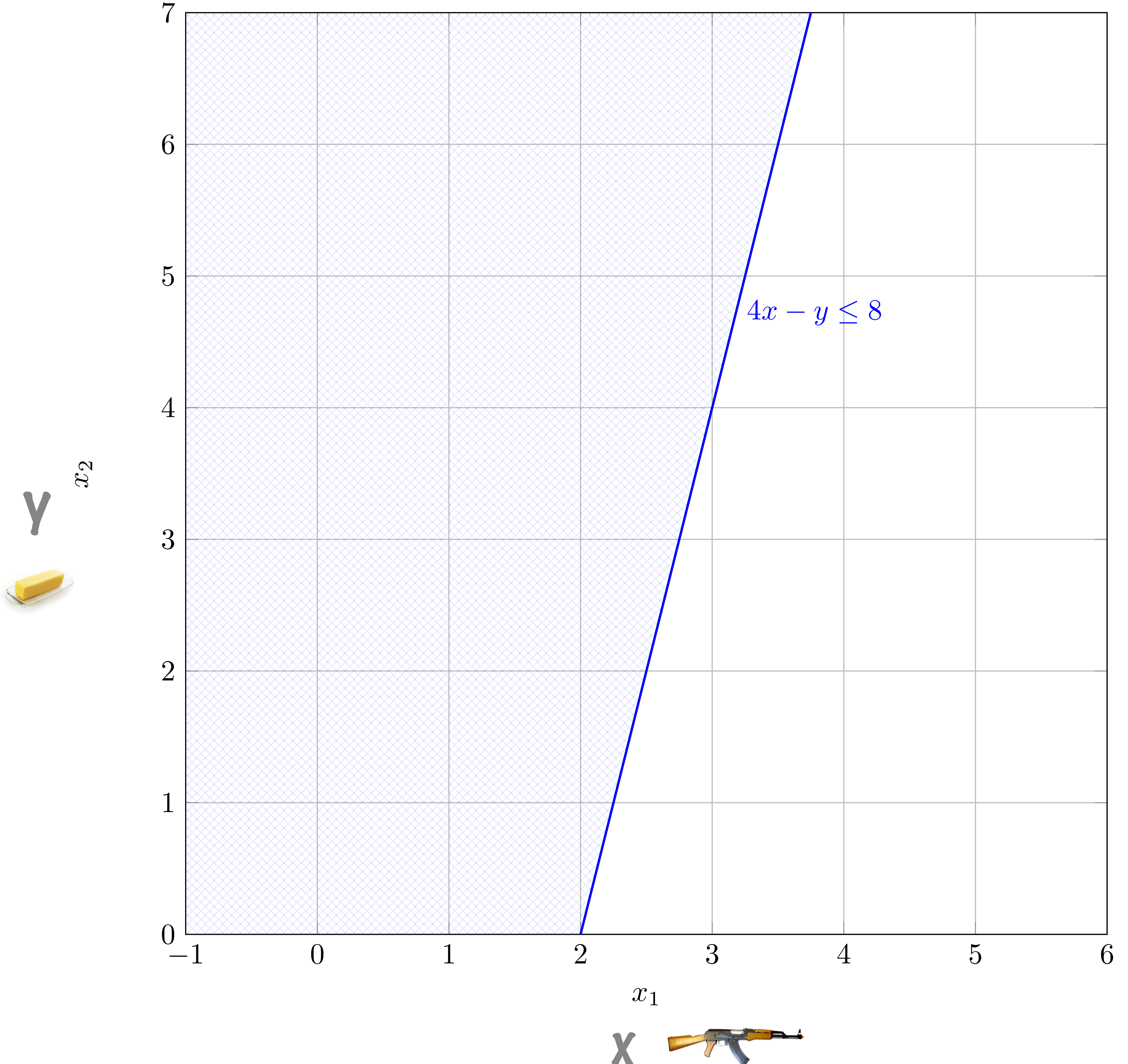
Guns and butter

$$\max x + y$$

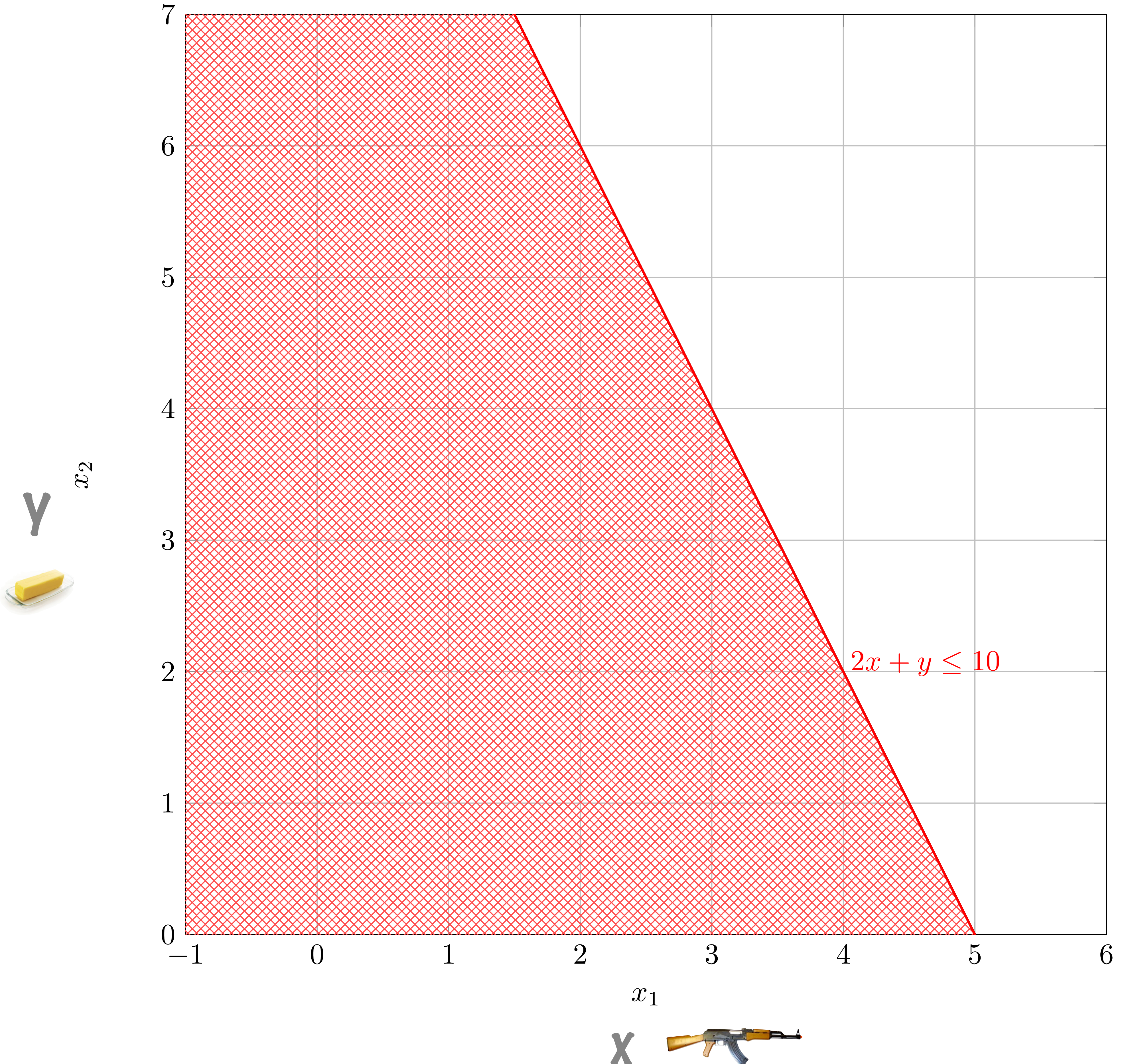
$$\begin{aligned} 4x - y &\leq 8 \\ 2x + y &\leq 10 \\ 5x - 2y &\geq -2 \\ x, y &\geq 0 \end{aligned}$$

http://i16.photobucket.com/albums/b20/safebuy/ak47/ak47-electric_lg.jpg

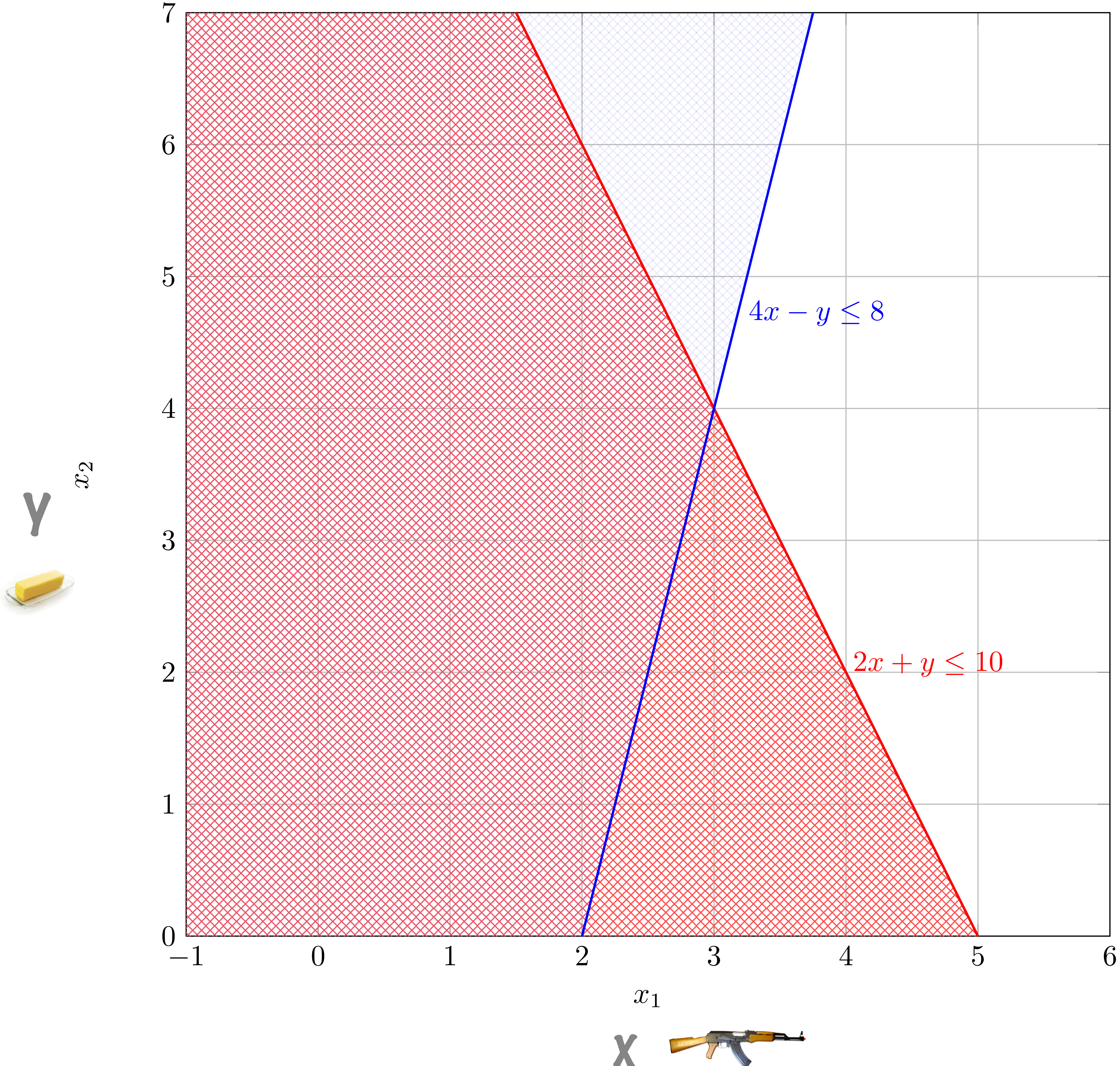
http://2.bp.blogspot.com/_NX4zcMNX4VE/Sb8MQffllI/AAAAAAAAAL0/eu4J0dfFhJE/s400/gourmet-butter.jpg



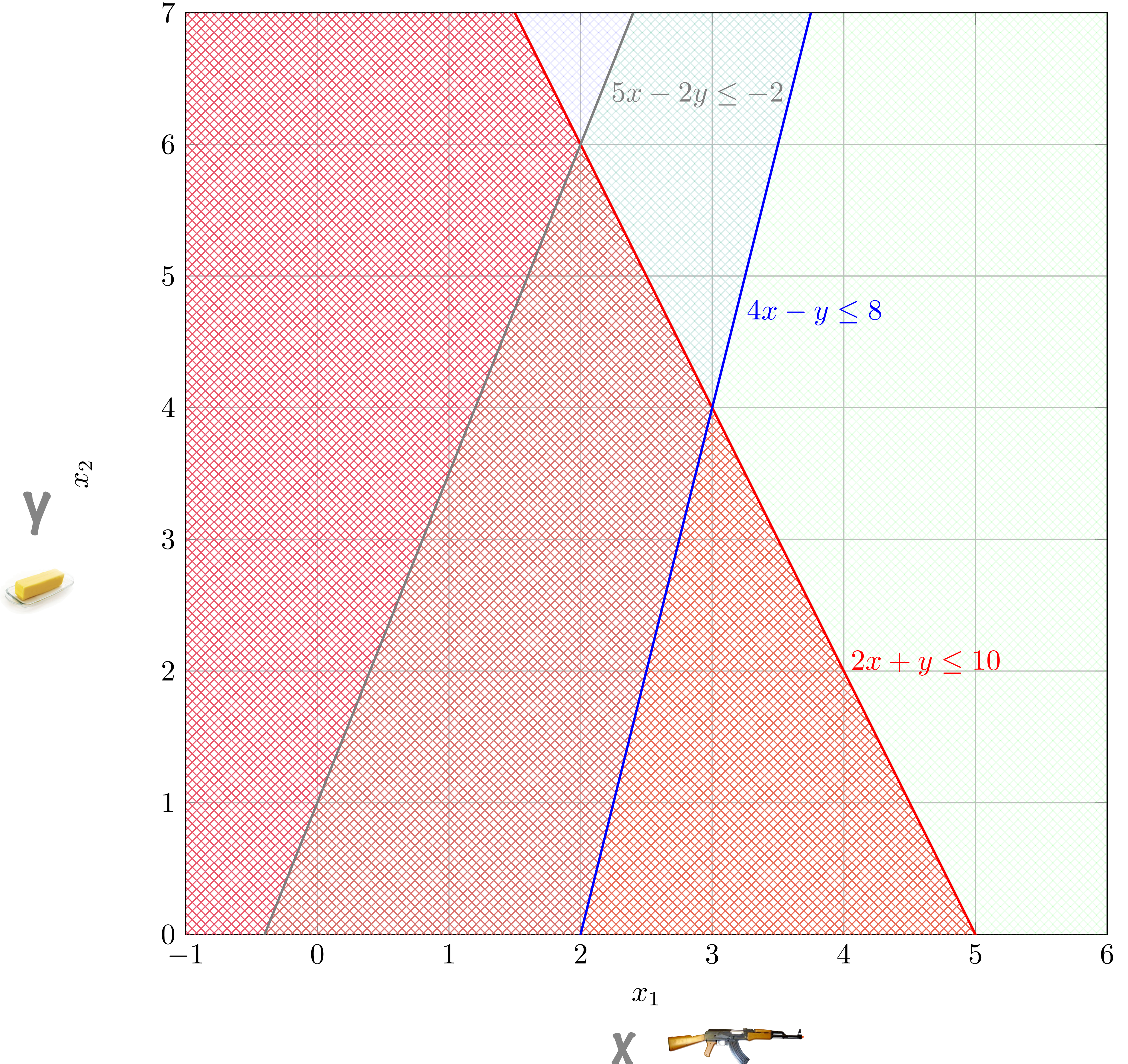
$$\begin{aligned} 4x - y &\leq 8 \\ 2x + y &\leq 10 \\ 5x - 2y &\geq -2 \\ x, y &\geq 0 \end{aligned}$$



$$\begin{aligned} 4x - y &\leq 8 \\ 2x + y &\leq 10 \\ 5x - 2y &\geq -2 \\ x, y &\geq 0 \end{aligned}$$



$$
\begin{aligned}
4x - y &\leq 8 \\
2x + y &\leq 10 \\
5x - 2y &\geq -2 \\
x, y &\geq 0
\end{aligned}
$$



$$
\begin{aligned}
4x - y &\leq 8 \\
2x + y &\leq 10 \\
5x - 2y &\geq -2 \\
x, y &\geq 0
\end{aligned}
$$

Certificate of optimality

$$\max x + y$$

$$\begin{aligned} 4x - y &\leq 8 \\ 2x + y &\leq 10 \\ 5x - 2y &\geq -2 \\ x, y &\geq 0 \end{aligned}$$