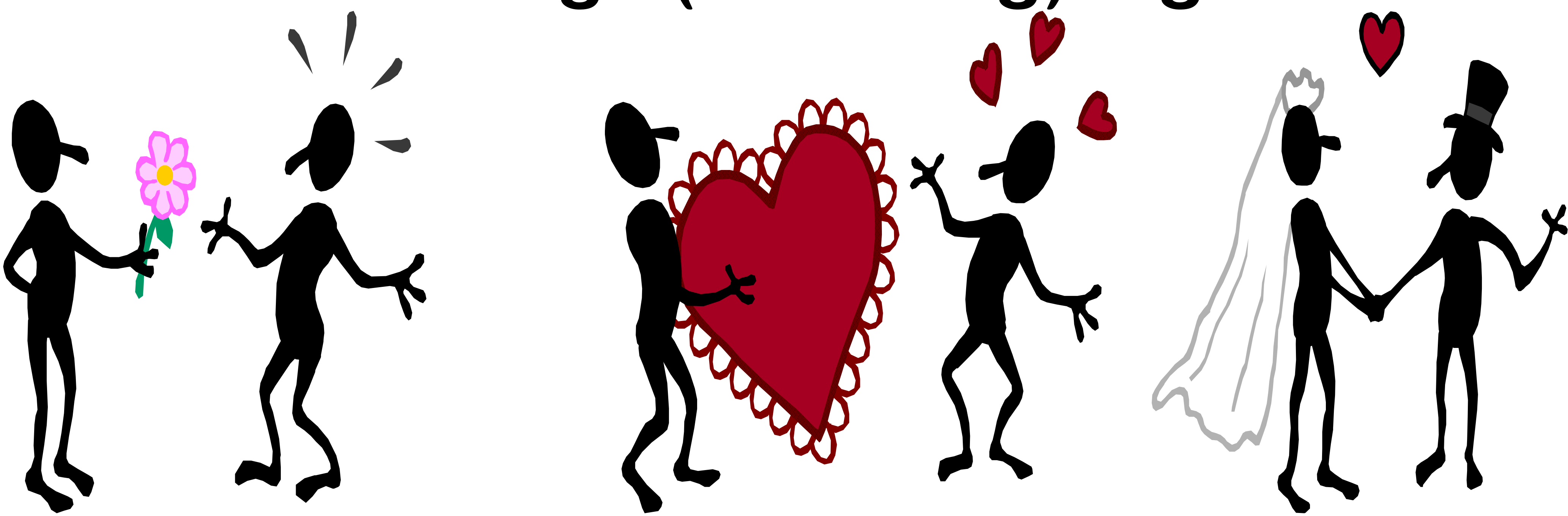
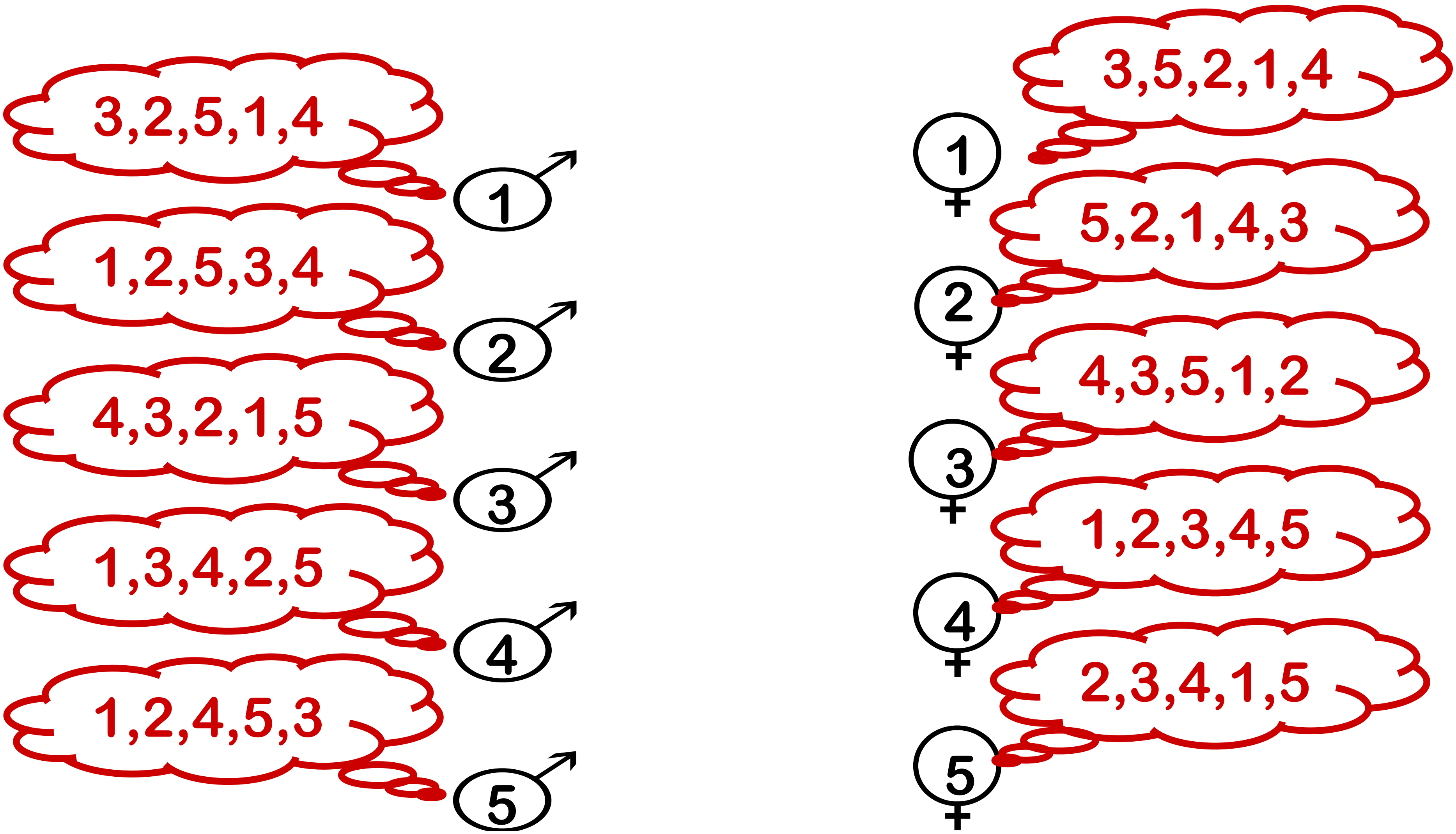
Credits:

Steven Rudich

The Mathematics Of 1950's Dating: Who wins the battle of the sexes? Stable marriage (matching) algorithm.



- There are *n* males and *n* females
- Each female has her own ranked preference list of all the males
  - E.g., women #1 most prefers male #3 over any other male.
- Each male has his own ranked preference list of the females
- How should we match them (1-to-1)



Product Details

- **Series:** Foundations of Computing

- **Hardcover:** 258 pages
- **Publisher:** The MIT Press (August 22, 1989)
- **Language:** English
- **ISBN-10:** 0262071185
- **ISBN-13:** 978-0262071185
- **Product Dimensions:** 9.4 x 7.3 x 0.8 inches
- **Shipping Weight:** 1.4 pounds

Definition of a Matching in this lecture

- A **matching** in this context is a list of couples that according to the algorithm, should be matched to each other. Each male is married to a single female and vice versa.

$M = \{ (m_1, f_{13}), (m_2, f_7), \ldots (m_n, f_n) \}$

The algorithm aims to find a good matching (under some definition)

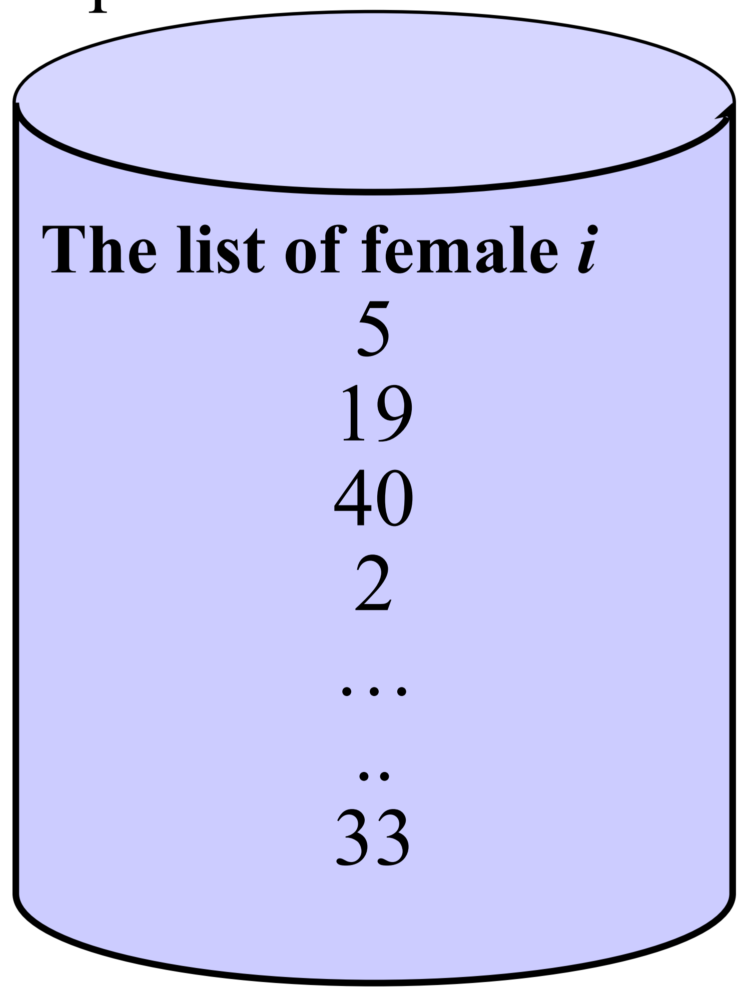
- Sometimes the term **pairing** is used

Definitions about the preference lists

In her list,

- male 5 is her **top choice.**
- If he is not interested, her top choice is male 19.
- If neither 5 nor 19 are interested, his top is 40 …

– This is a full ranking of all males.



Definition: Rogue Couples

•Consider a given matching M (that is, assume that matching is done) .

A rouge couple (in this matching) is a couple (female, male) who are **not** married to each other, but prefer each other over their spouses.

•In the example to the right

Zod is married to **Evanora (6)**, but prefers **Aradia (3)**

Aradia is married to **Syndrom (5)** ,but prefers **Zod (2)**

Zod's list

----

1. Allegra
2. Beatrix
3. **Aradia**
4. Cassandra
5. Cordelia
6. **Evanora**
7. Gullveig

Aradia's list

----

1. Mr Burn
2. **Zod**
3. Hannibal
4. …
5. **Syndrom**
6. …
7. Gus Fring
8.

•They will be called a <u>rogue couple</u>.

•They both would gain from dumping their mates and marry each other.

•A source of confusion: A couple that is married to each other could not be rouge. The other couples are the ones we are concern about.

•A matching is called **stable** if it does not contain any rouge couples.

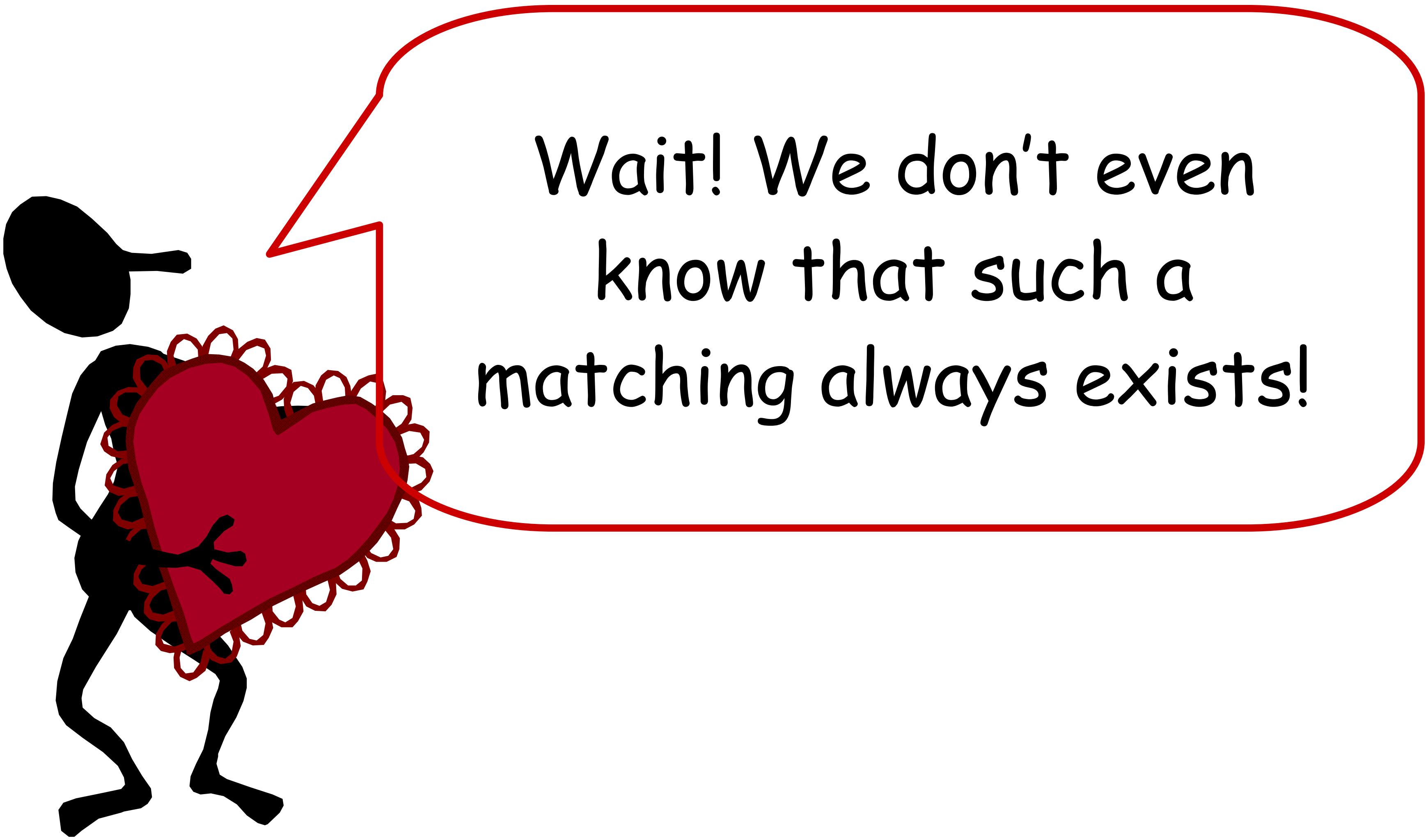
•The source of the 'instability': They would **both** benefit from changing the situation

•How could we obtain stability: Make sure that if one gains, the other loose

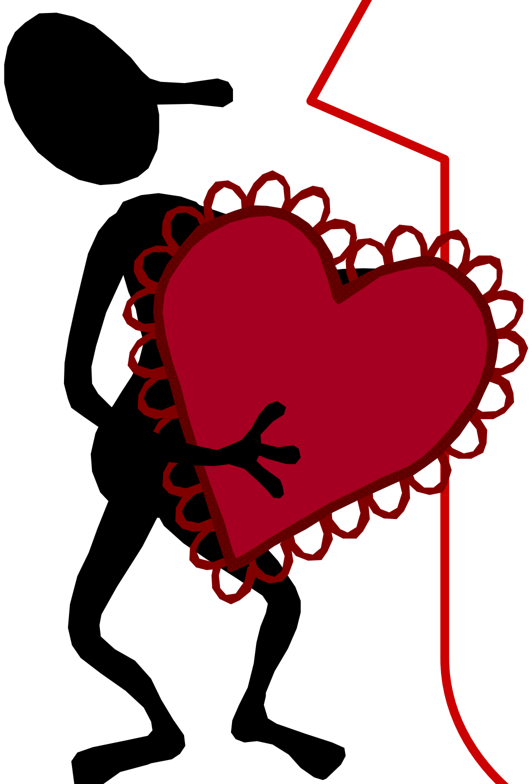
The study of stability will be the subject of the entire lecture.

We will: Analyze various mathematical properties of an algorithm that looks a lot like 1950's dating.

Given a set of preference lists, how do we find a stable matching?

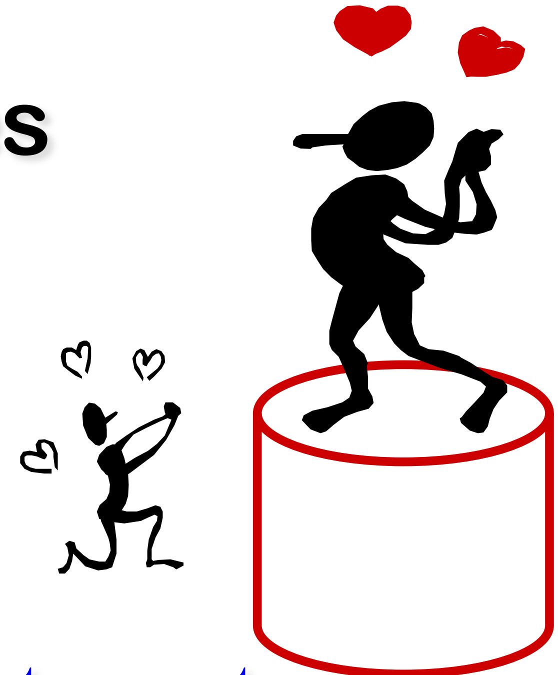


Is there always a stable matching ?



- Will show: <u>every</u> set of preference lists have a stable matching.
- Will prove it by presenting a fast algorithm that, given any set of input lists, will output a stable matching.

Terminology and principles



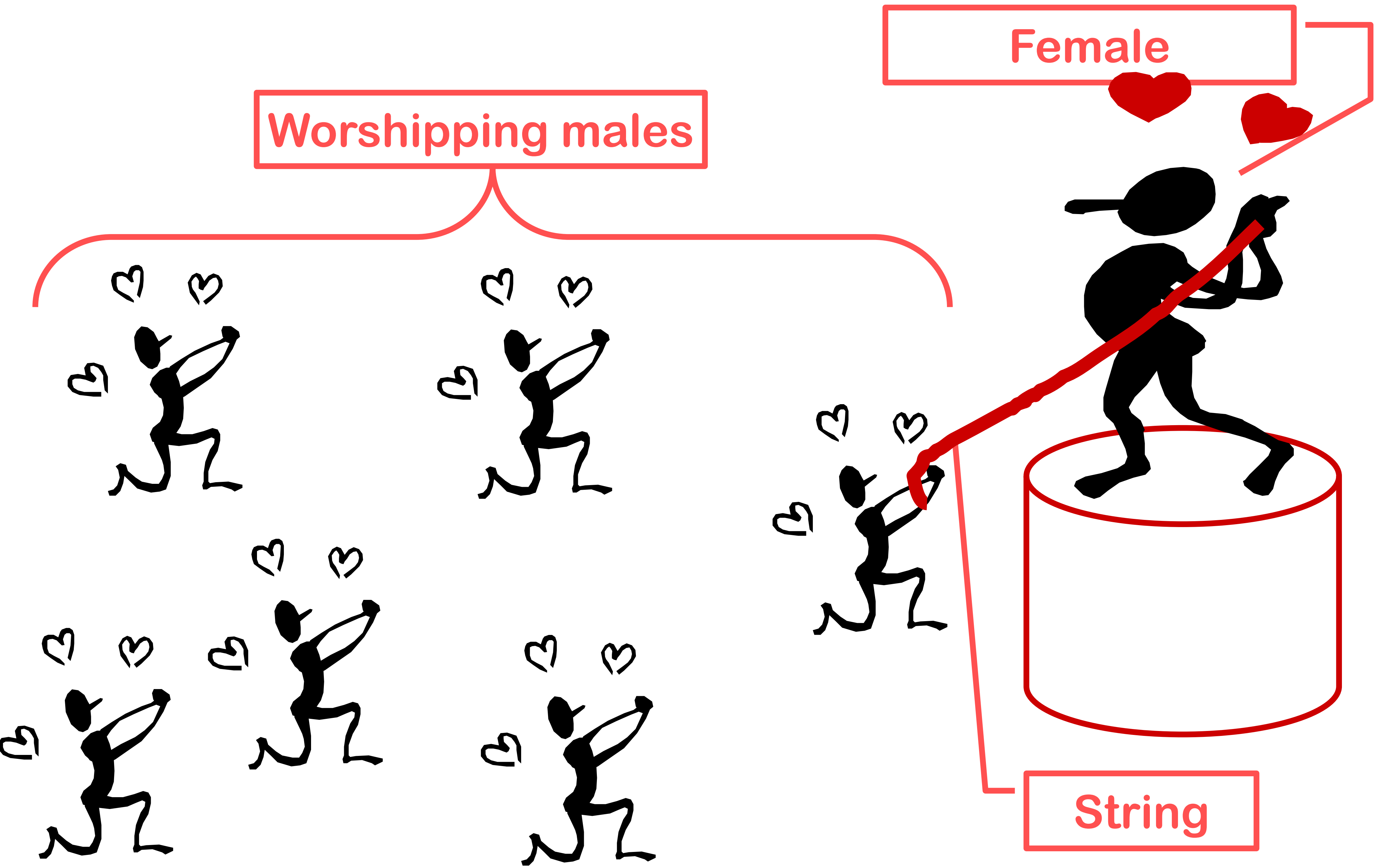
- A male can **propose** (marriage) to a female.
- A female can **reject** the proposal.

- During most of the process, a female would not accept a proposal, but would tell a proposing male "**maybe**".
- This is called "**putting the male on a string**"

- Once a male is rejected, he **crosses** off from his list the rejecting female – he will not propose to her again.

- Once a male proposes, he cannot change his mind until he is rejected.

The Traditional Marriage Algorithm



Traditional Marriage Algorithm (TMA)

1) Repeat at each day {

- Morning
  - Each male proposes to the best female whom he has not yet crossed off
- Afternoon (for each females with at least one proposal)
  - To today's best offer: **"Maybe, come back tomorrow" (putting him on a string)**
  - All other proposals are rejected.
- Evening
  - Any rejected male crosses the rejecting female off his list.

}Until all males are on strings.

2) Each female marries the last male she just said "maybe"

Note: Each male proposes to females in decreasing order on his list.

<u>Lemma</u>: If a female has a male b on a string, then she will either marry him, or marry someone she prefers over him.

Proof:

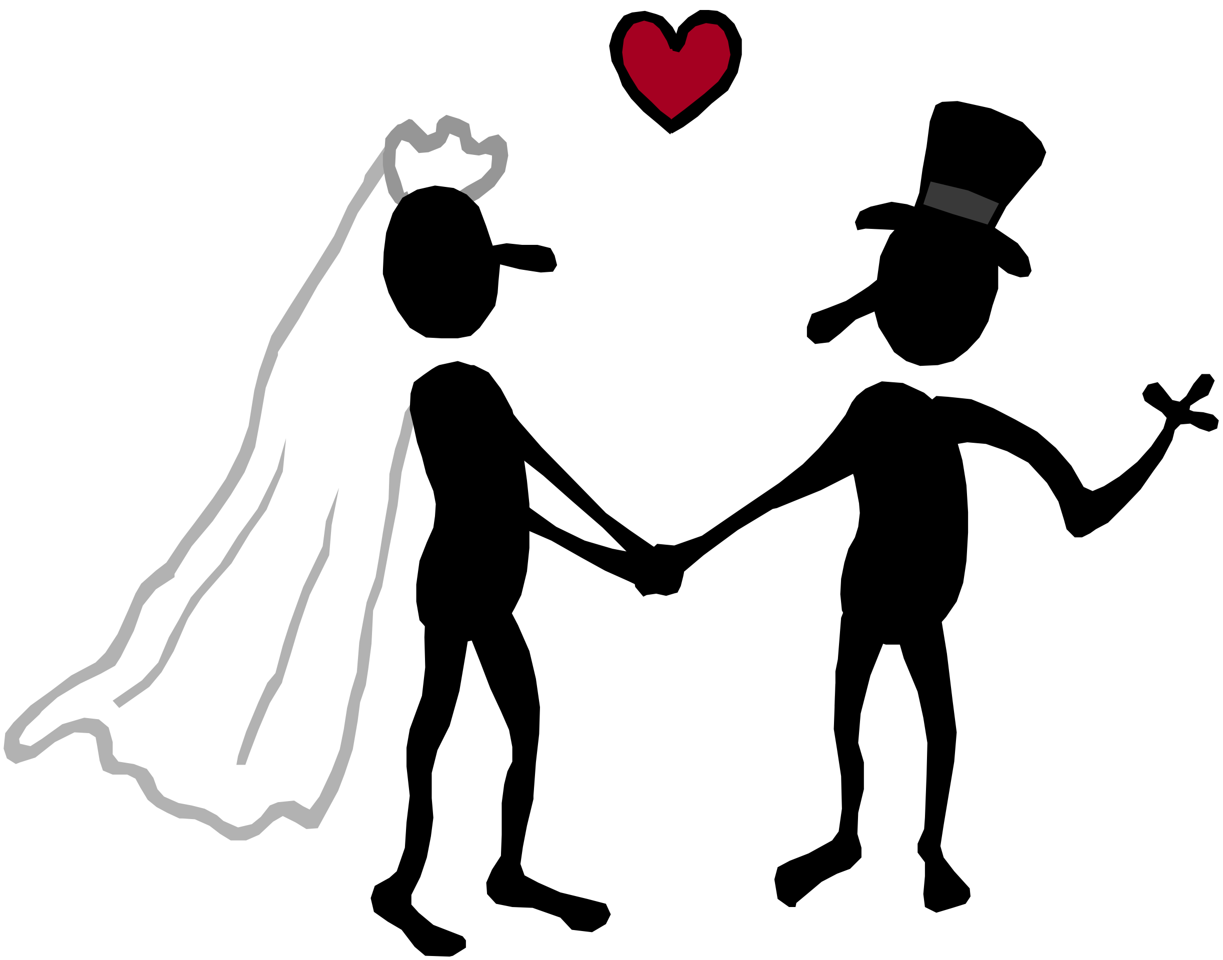
- She would only let go of *b* in order to "maybe" b′ which she prefers over *b*
- She would only let go of b′ for someone b″ she prefers over b′ etc.

When the process terminates, she is left with someone she prefers over b.

QED

Corollary:

Each female will marry her absolute favorite of the males who visit her during the Traditional Marriage Algorithm (TMA)



<u>Lemma</u>: No male can be rejected by all the females

- Proof by contradiction.
- Suppose male *b* is rejected by all the females. At that point:
  - Each female must have a suitor other than ***b*** (By previous Lemma, once a female has a suitor she will always have at least one)
  - The ***n*** females have ***n*** suitors, ***b*** not among them. Thus, there are at least ***n+1*** males.

  Contradiction

  QED

Theorem:
The TMA always terminates after at most n^2 days

Proof

– The total length of the lists of all males is

$$n \times n = n^2.$$

–Each day at least one male gets a “No”, so at least one female is deleted from one of the lists.

–Therefore, the number of days is bounded by the original size of the master list $= n^2$.

QED

Great! We know that TMA will terminate and produce a pairing.

But is it stable?

Theorem: TMA. Produces a stable pairing.

- Let m_1 and f_1 be any couple in T.
- Suppose m_1 prefers f_2 over f_1 .
- We will argue that f_2 prefers her husband over m_1 .
- During TMA, male m_1 proposed to f_2 before he proposed to f_1 .
- Hence, at some point f_2 rejected m_1 for someone she preferred.
- By the Improvement lemma, the man she married was also preferable to m_1
- Thus, every male will be rejected by any female he prefers to his wife.
- T is stable. QED.

Uniqueness

- Question: Given the input preference lists, is there only one stable matching ?

- Answer: Sometimes yes, sometime no. Depending on the input.

- Remember TMA produces a stable matching, but not every possible matching is an output of TMA.

A slow algorithm to produce every stable matching

- For every possible matching M
  - Check if M is stable.

Happiness / Optimality

- Assume a pairing is stable (under the definition given a few slides ago)
- Question: Will every person get her/his top choice in every stable ?

- Answer: Not necessarily (see whiteboard)

- So could we claim that it is minimize or maximize something?
- ANSWER: No. But lest understand why.

The Optimal female

Consider a male m_i (say "Mark").

Definition We say that a female f_j is the optimal female for Mark if f_j is highest ranked female (in Mark's list) for whom there is some stable matching in which Mark is married to f_j.

She is the best female he can get in a stable world.

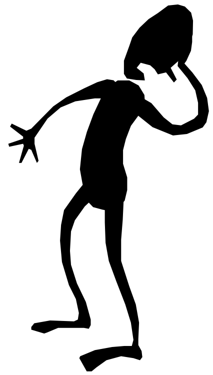
- Presumably, she might be better than the female he gets in the stable pairing output by TMA.
- Note – she might or might NOT be the highest female on his list).
- And note that different males **might** have different opt females.

Thm

- The Traditional Marriage Algorithm yields a matching at which each male gets his optimal female

- That is, TMA produces a a male-optimal pairing

Next slides will be dedicated to prove this Theorem



TMA but with exact clock.

- Assume: At each time stamp, (every `tick' of the clock) there is exactly one **event:**
  - Event: a single man proposes, and if got rejected, his next proposal will be in next time stamp)
- Note: The exact order is not crucial:
  - If both m_1, m_2 are proposing to f, the result is the same independent of whom proposed first.

Proof of Thm:
(TMA produces a male-optimal pairing)

We will show that no male is being rejected by his **opt** female.

- Suppose by contradiction that **Florence** is the **opt** female of **Adam**, yet during the TMA she has just rejected him to maybe **Bob.**
- Assume this is the first time a male is rejected by his **opt** female.

(some males might have been rejected, but not by their opts)

- **Bob** had not yet been rejected by his optimal female

Thm: TMA produces a male-optimal pairing (cont)

- **Bob** had not yet been rejected by his optimal female. Therefore in **Bob's** list
  Florence is either **Bob's** **optimal** female. Or
  Florence is higher than his Bob's optimal.

That is, in any stable world, **BOB** is either married to **Florence**, or to somebody lower on his list *(definition of opt)*

- Let ***S*** be the matching at which (**Adam**. **Florence*)*** are married

(***S*** is NOT the result of the TMA). Think about S as a parallel universe.

- In ***S*,** **Bob** is married to f_2 , whom he prefers less than **Florence.**
- **hence** (**Bob, Florence)** are a **rouge** couple, so ***S*** is unstable

QED

The Pessimal male

- Let **Florence** be one of the females.
- Note that there might be different matching which are stable.

- **Florence's** pessimal male is the lowest ranked male (**on her list**) for whom there is some stable matching at which she gets him.

- He is the worst male she can conceivably get in a stable world.

Thm: The TMA is female-pessimal.

Proof: We know TMA it is male-optimal.

(**Syndrome**, **Florence**) is a couple in TMA,

➔**Florence** is **Syndrome's** optimal female (he cannot do better in a stable world) (the previous theorem)

Consider a stable matching ***S*** where **Florence** does worse than **Syndrome.**

Let **Zod** be **Florance's** husband in ***S***

(Zod is lower on her list than Syndrome)

In ***S,*** **Syndrome** is not married **Florence**, (taken by **Zod**) and cannot marry a female he prefers over **Florence** (otherwise she is not his opt).

So (**Syndrome** , **Florence**) is a rogue couple.

– **Therefore, *S* is not stable. QED**

REFERENCES

- D. Gale and L. S. Shapley, College admissions and the stability of marriage, American Mathematical Monthly 69 (1962), 9-15

- Dan Gusfield and Robert W. Irving, The Stable Marriage Problem: Structures and Algorithms, MIT Press, 1989