



Kapitel 5: Sortieren

*Algorithmen und Datenstrukturen
WS 2022/23*

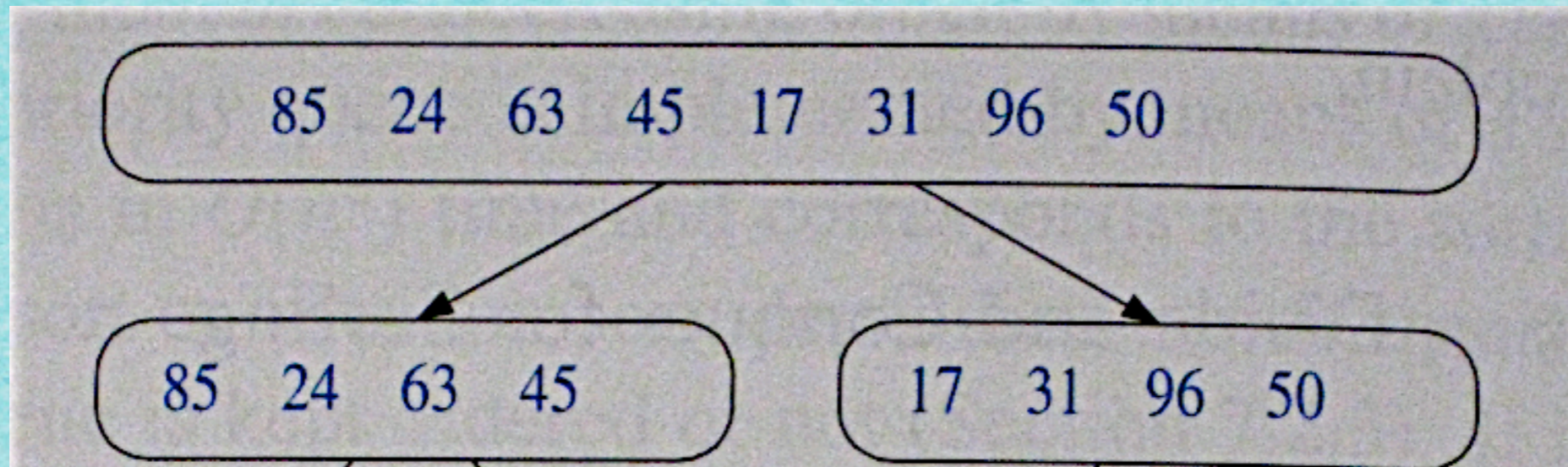
Prof. Dr. Sándor Fekete

5.1 Mergesort

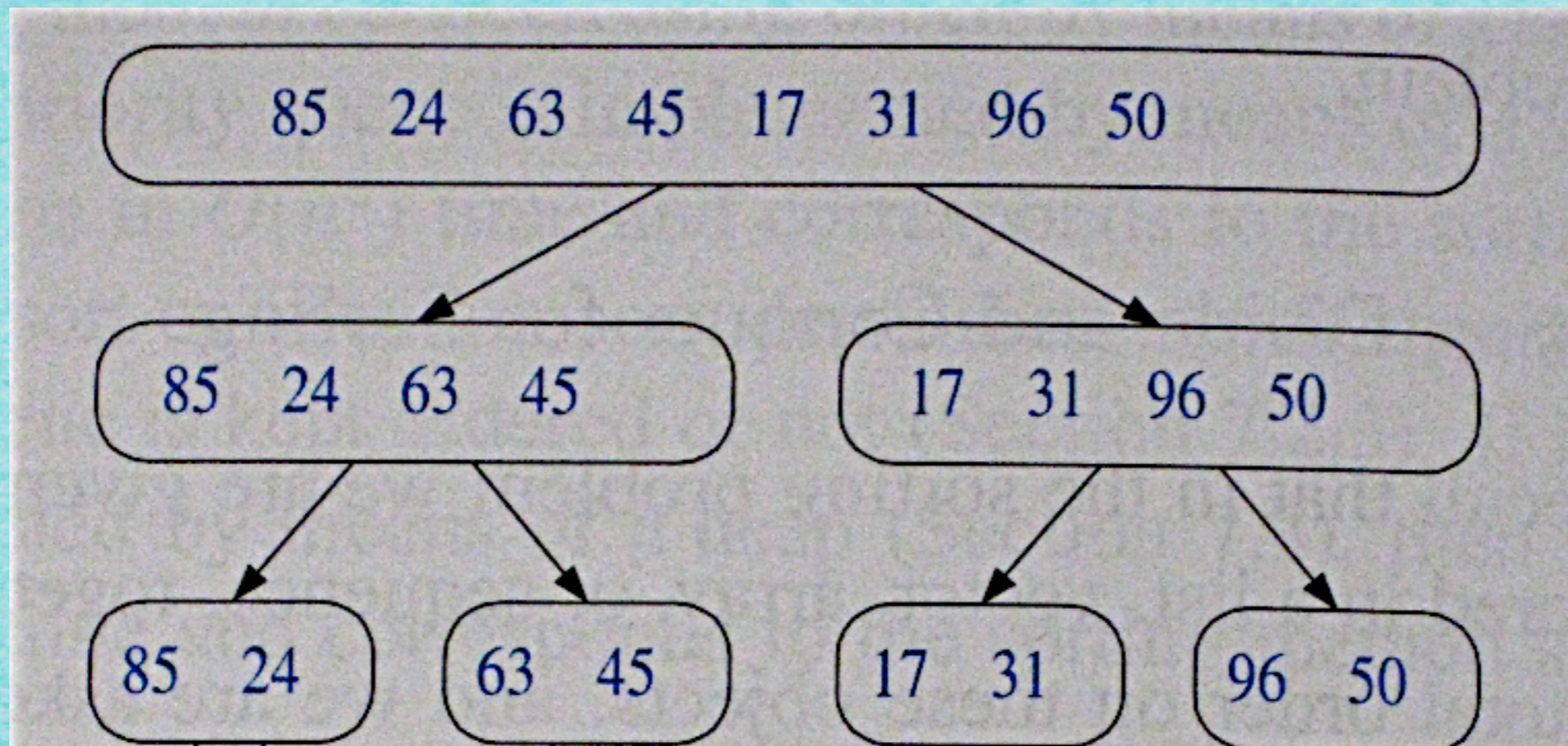
5.1 Mergesort

85 24 63 45 17 31 96 50

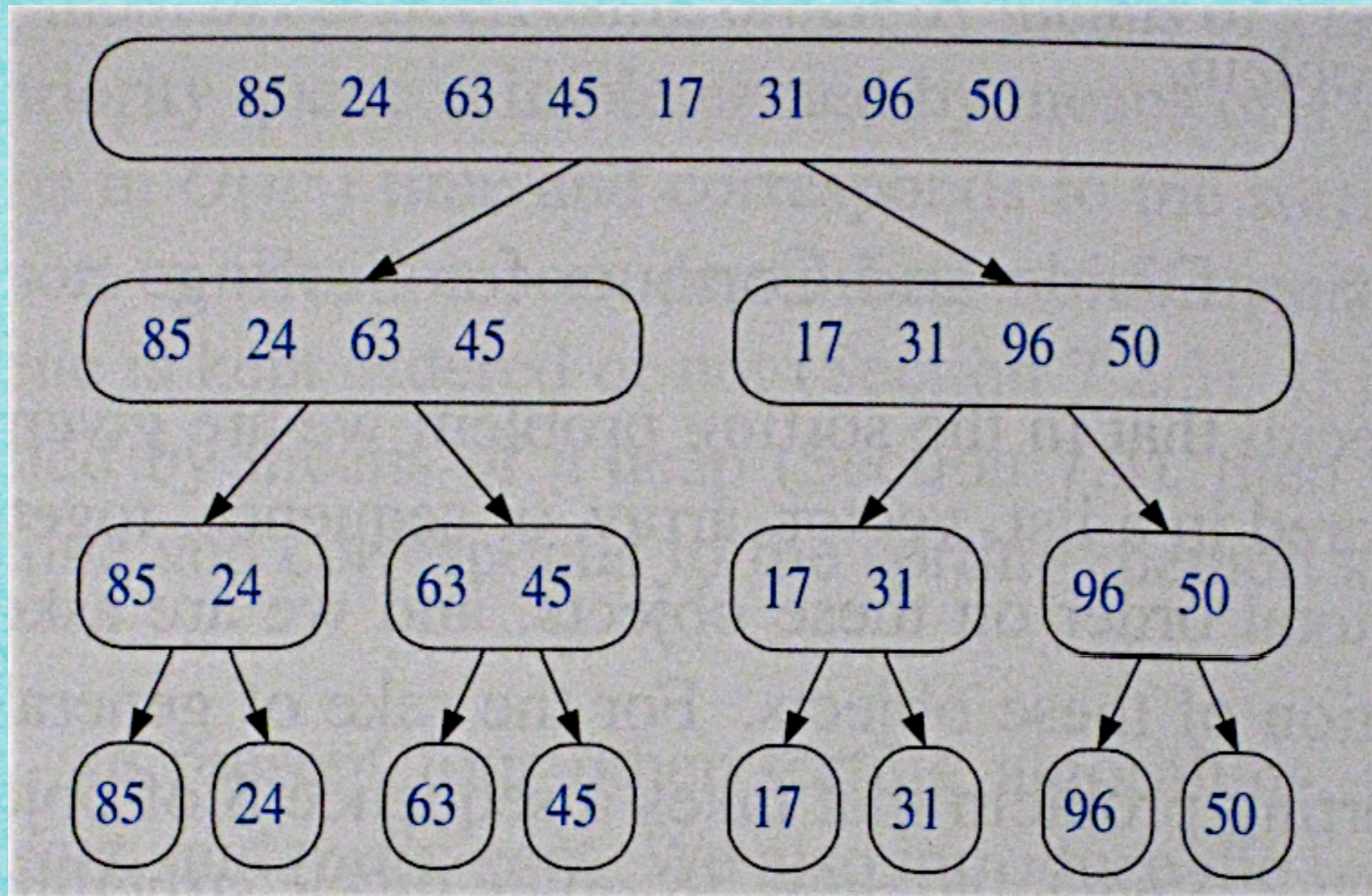
5.1 Mergesort



5.1 Mergesort



5.1 Mergesort

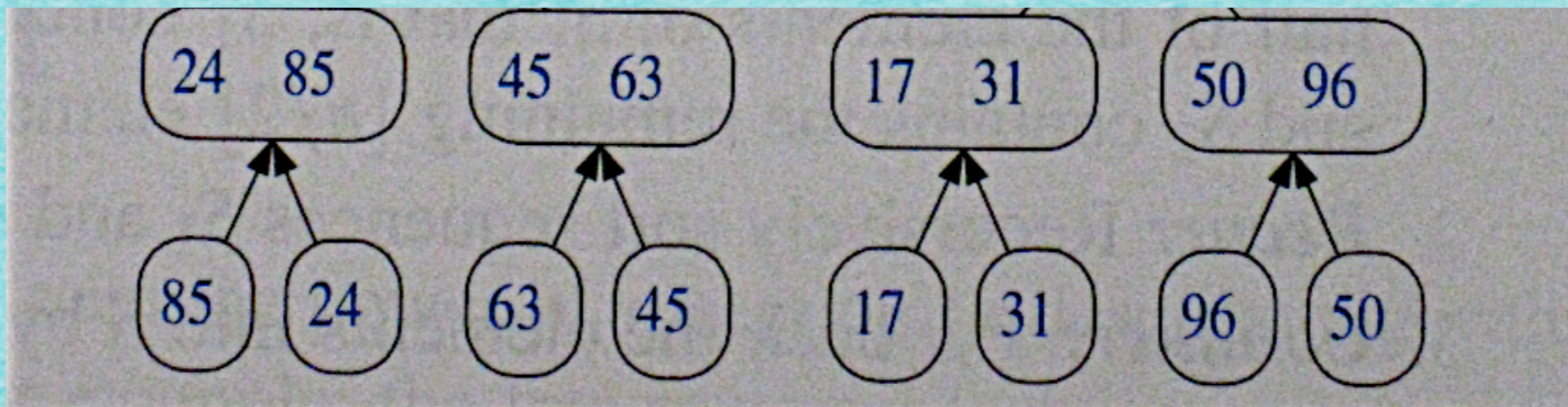


5.1 Mergesort

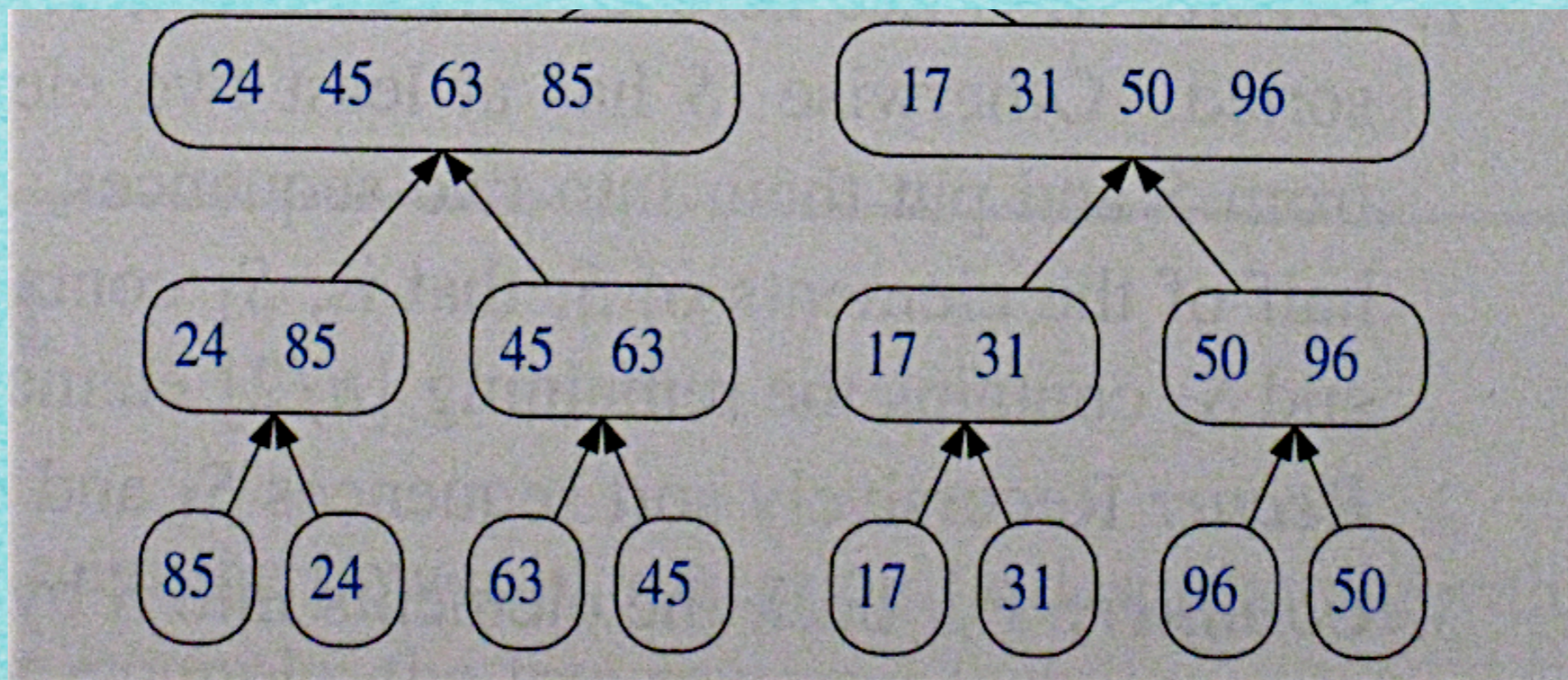
5.1 Mergesort

85 24 63 45 17 31 96 50

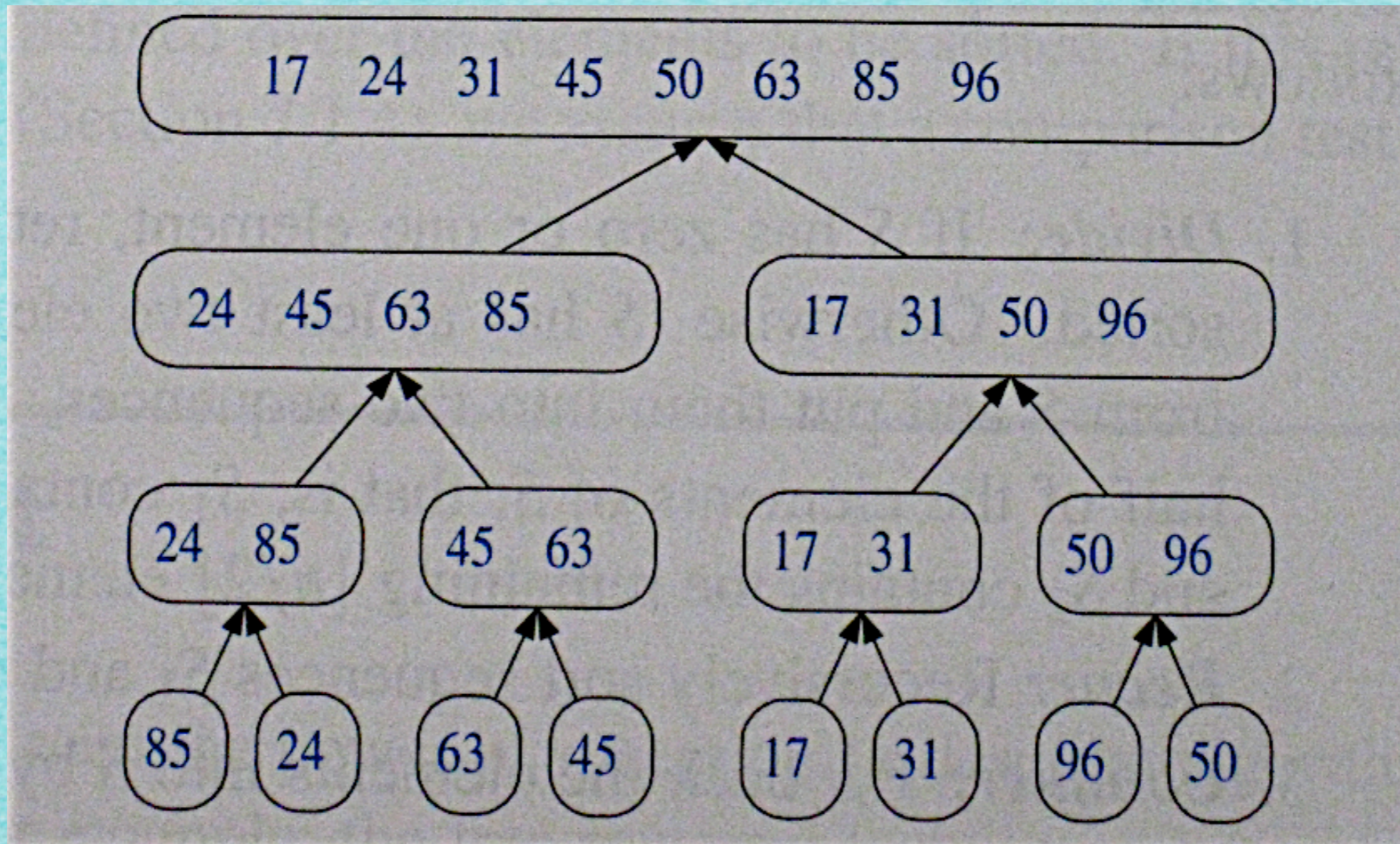
5.1 Mergesort



5.1 Mergesort



5.1 Mergesort



5.1.2 Algorithmische Beschreibung

Algorithmus 5.1

INPUT: Subarray von $A=[1,\dots,n]$,
der bei Index p beginnt und bei Index r endet, d.h. $A[p,\dots,r]$

OUTPUT: Sortierter Subarray

MERGE-SORT(A,p,r)

```
1  if  $p < r$ 
2    then  $q \leftarrow \lfloor (p + r)/2 \rfloor$ 
3         MERGE-SORT( $A, p, q$ )
4         MERGE-SORT( $A, q + 1, r$ )
5         MERGE( $A, p, q, r$ )
```


5.1.2 Algorithmische Beschreibung

Algorithmus 5.1

INPUT: Subarray von $A=[1,\dots,n]$,
der bei Index p beginnt und bei Index r endet, d.h. $A[p,\dots,r]$

OUTPUT: Sortierter Subarray

MERGE-SORT(A,p,r)

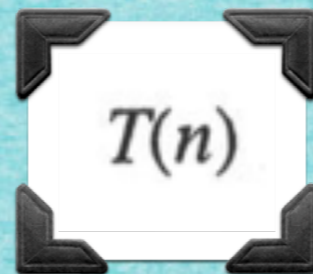
```
1  if  $p < r$ 
2    then  $q \leftarrow \lfloor (p + r)/2 \rfloor$ 
3         MERGE-SORT( $A, p, q$ )
4         MERGE-SORT( $A, q + 1, r$ )
5         MERGE( $A, p, q, r$ )
```


5.1.3 Laufzeit von Mergesort

Wie viele Schritte benötigt Merge-Sort für einen Array der Länge n ?

5.1.3 Laufzeit von Mergesort

Wie viele Schritte benötigt Merge-Sort für einen Array der Länge n ?



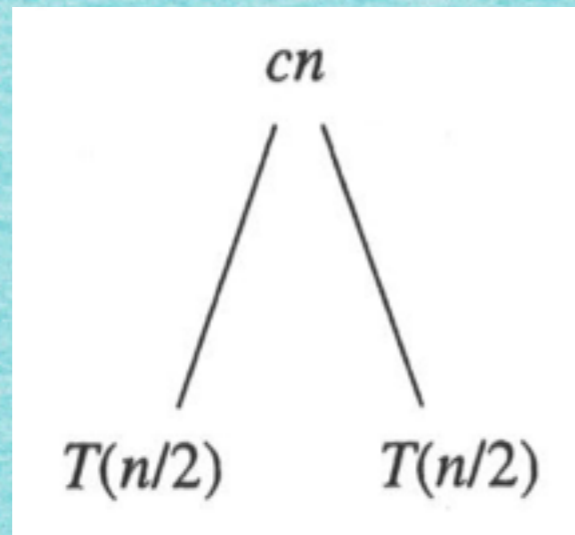
5.1.3 Laufzeit von Mergesort

5.1.3 Laufzeit von Mergesort

$T(n)$

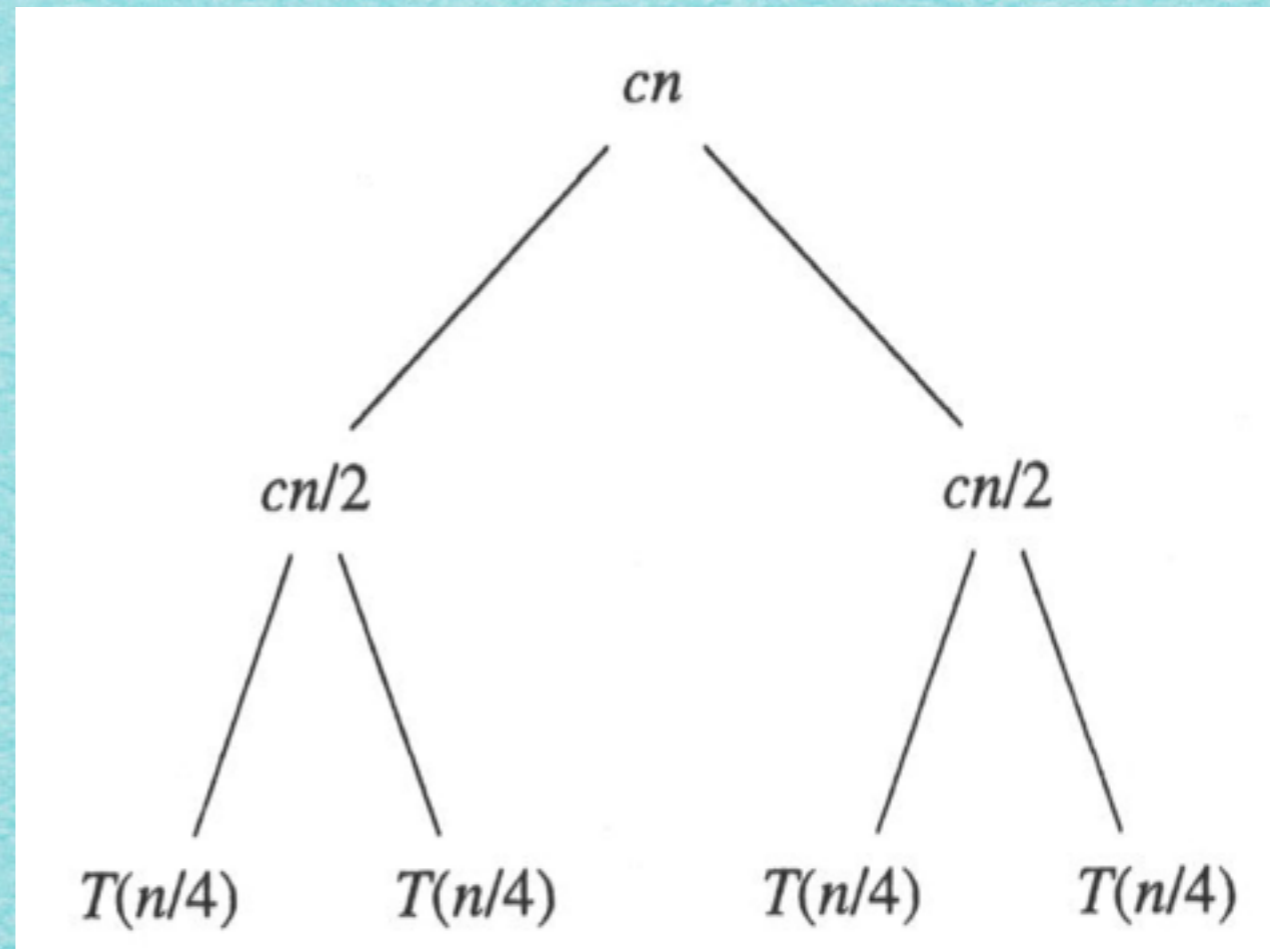
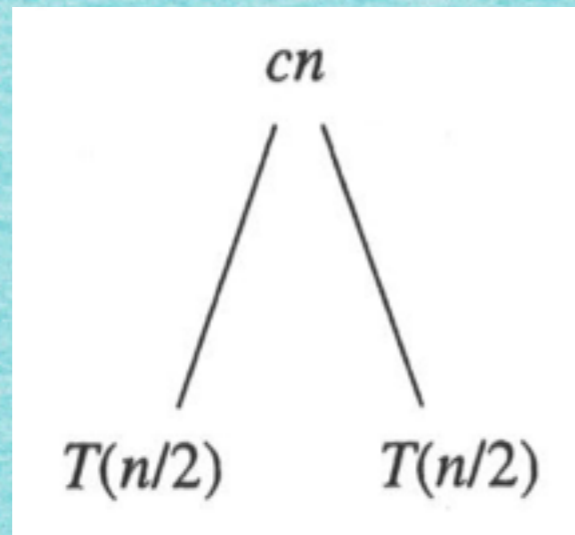
5.1.3 Laufzeit von Mergesort

$T(n)$

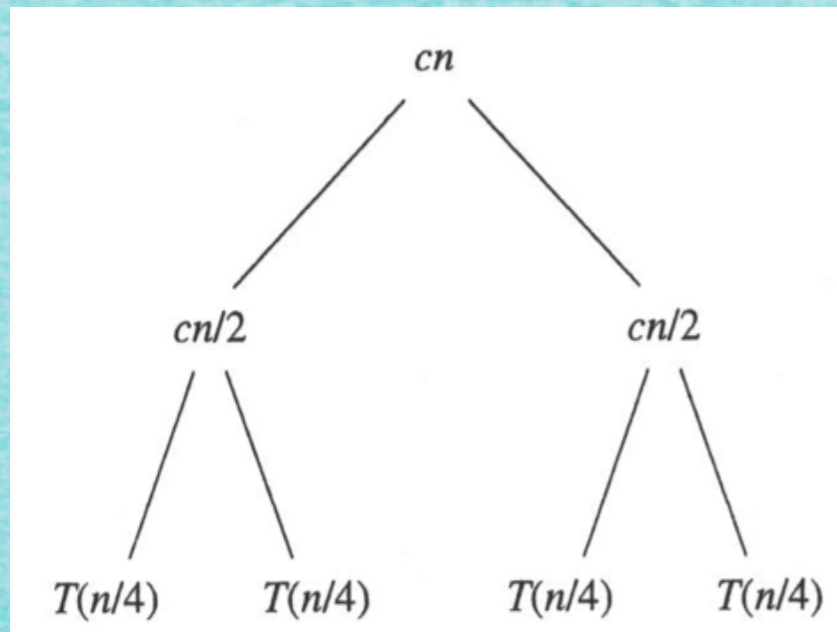


5.1.3 Laufzeit von Mergesort

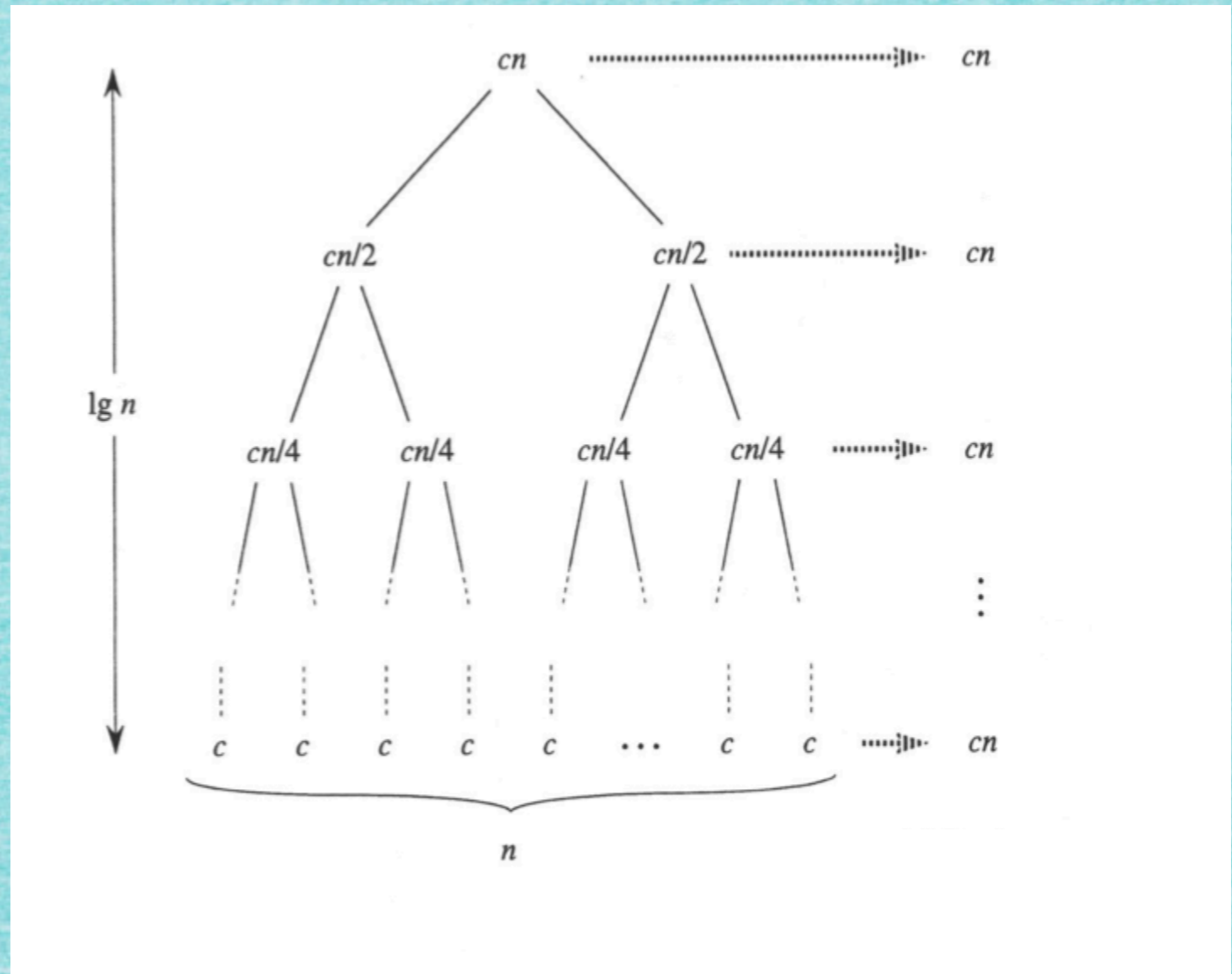
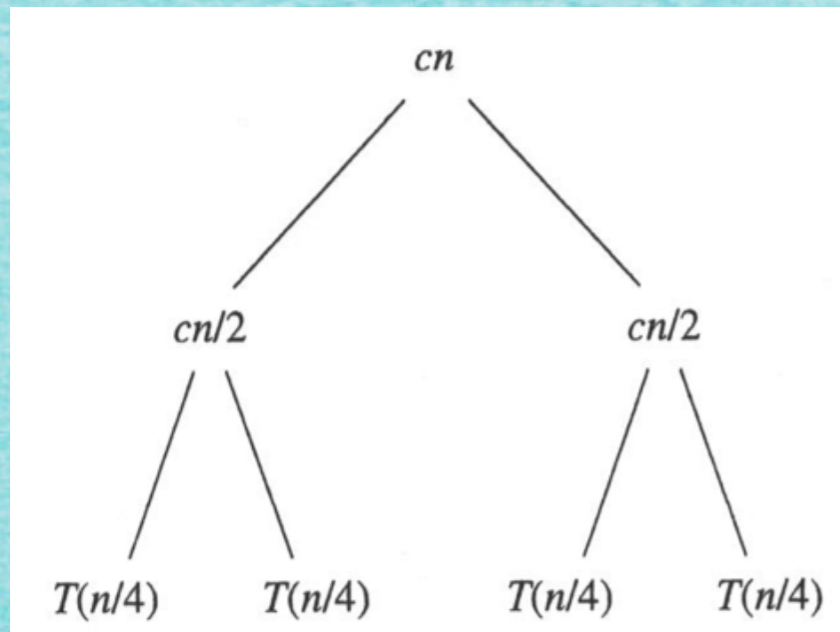
$T(n)$



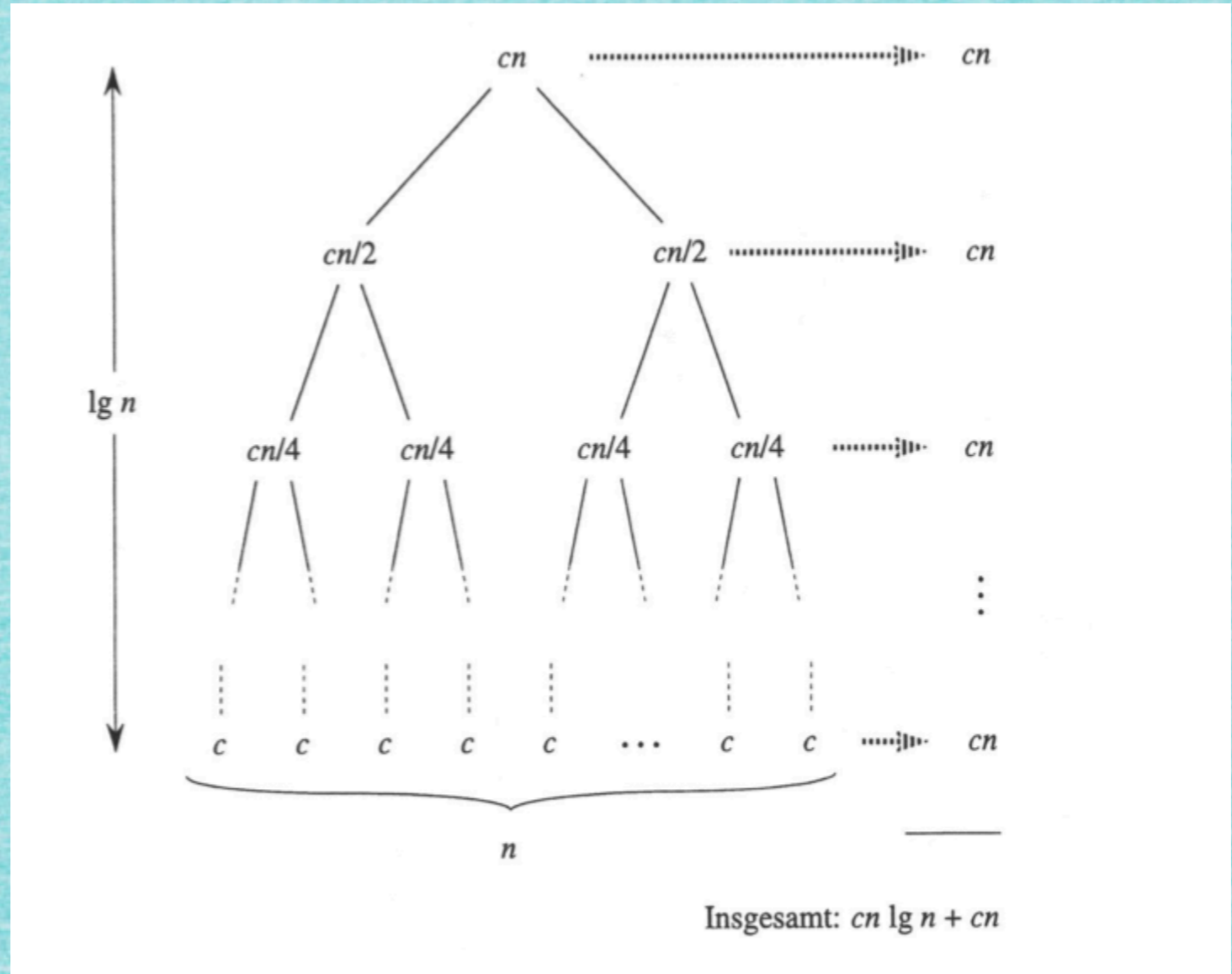
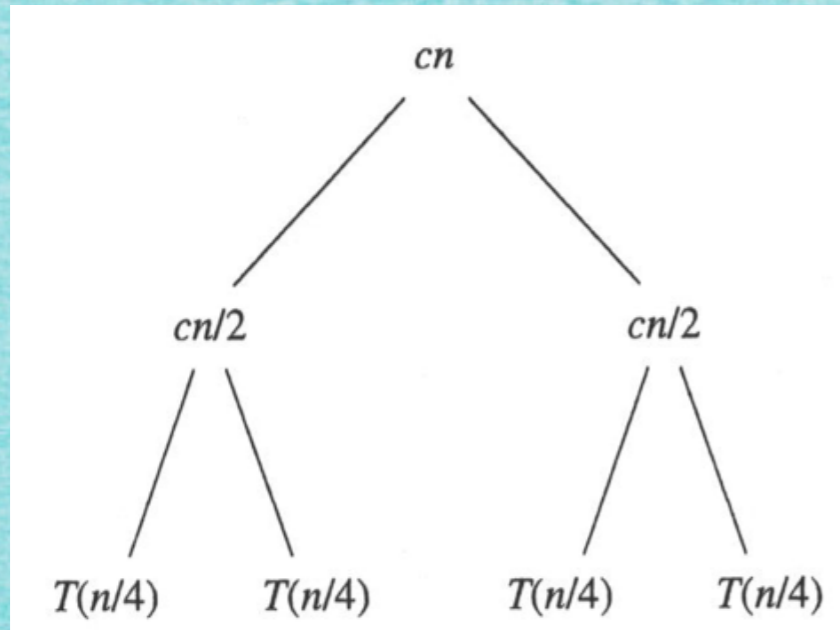
5.1.3 Laufzeit von Mergesort



5.1.3 Laufzeit von Mergesort



5.1.3 Laufzeit von Mergesort



5.1.3 Laufzeit von Mergesort

5.1.3 Laufzeit von Mergesort

Satz 5.3 (Komplexität von Mergesort)

5.1.3 Laufzeit von Mergesort

Satz 5.3 (Komplexität von Mergesort)
Für einen n -elementigen Array A hat Mergesort eine Laufzeit von $O(n \log n)$.

5.1.3 Laufzeit von Mergesort

5.1.3 Laufzeit von Mergesort

Fragen:

5.1.3 Laufzeit von Mergesort

Fragen:

- Geht's noch schneller?

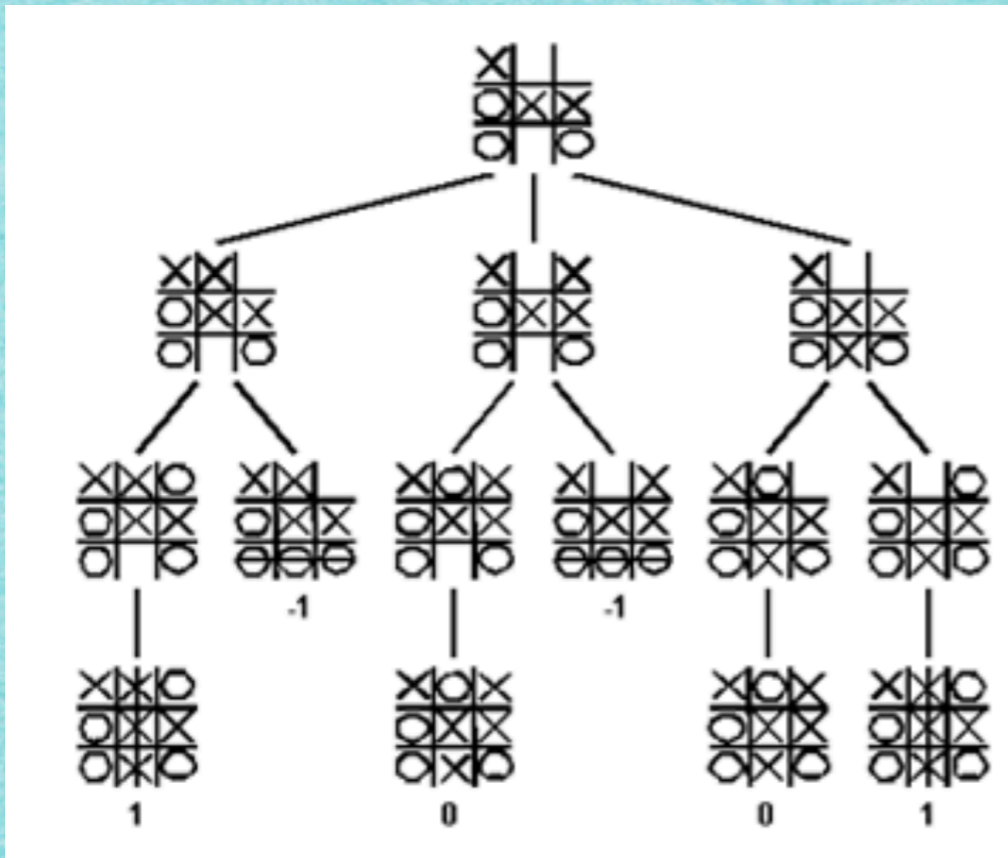
5.1.3 Laufzeit von Mergesort

Fragen:

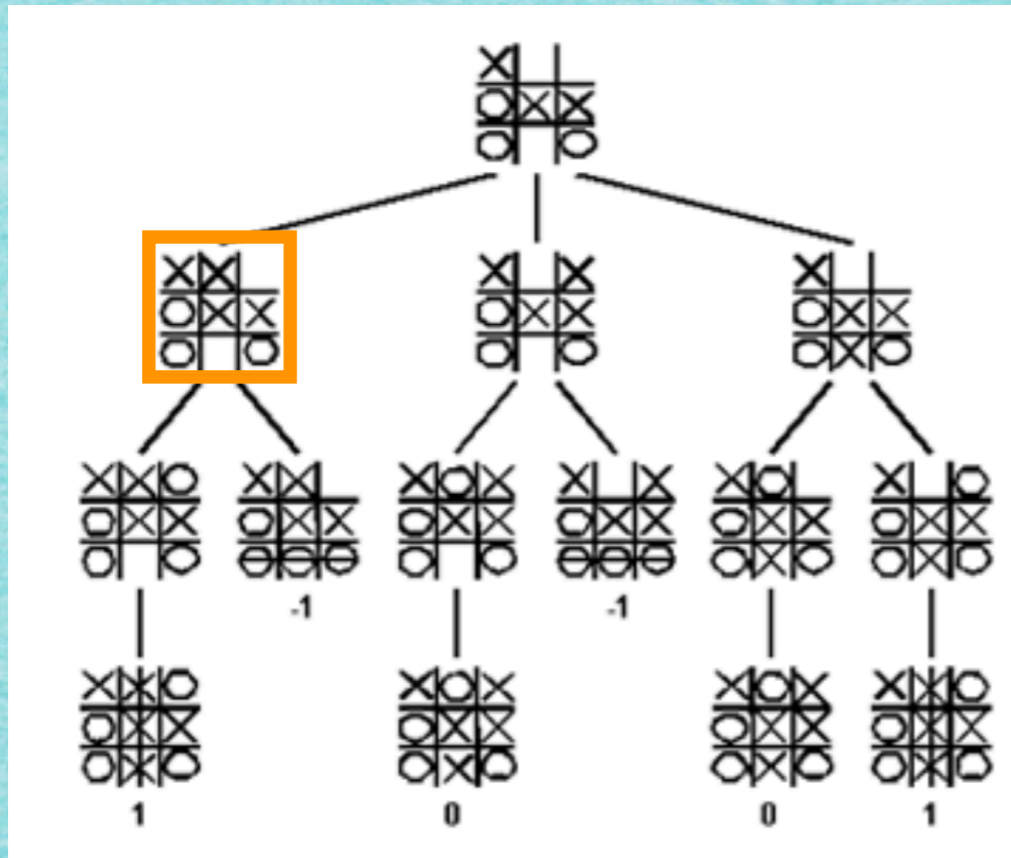
- Geht's noch schneller?
- Wie kann man sonst mit Rekursionen umgehen?

5.2 Schranken: Entscheidungsbäume

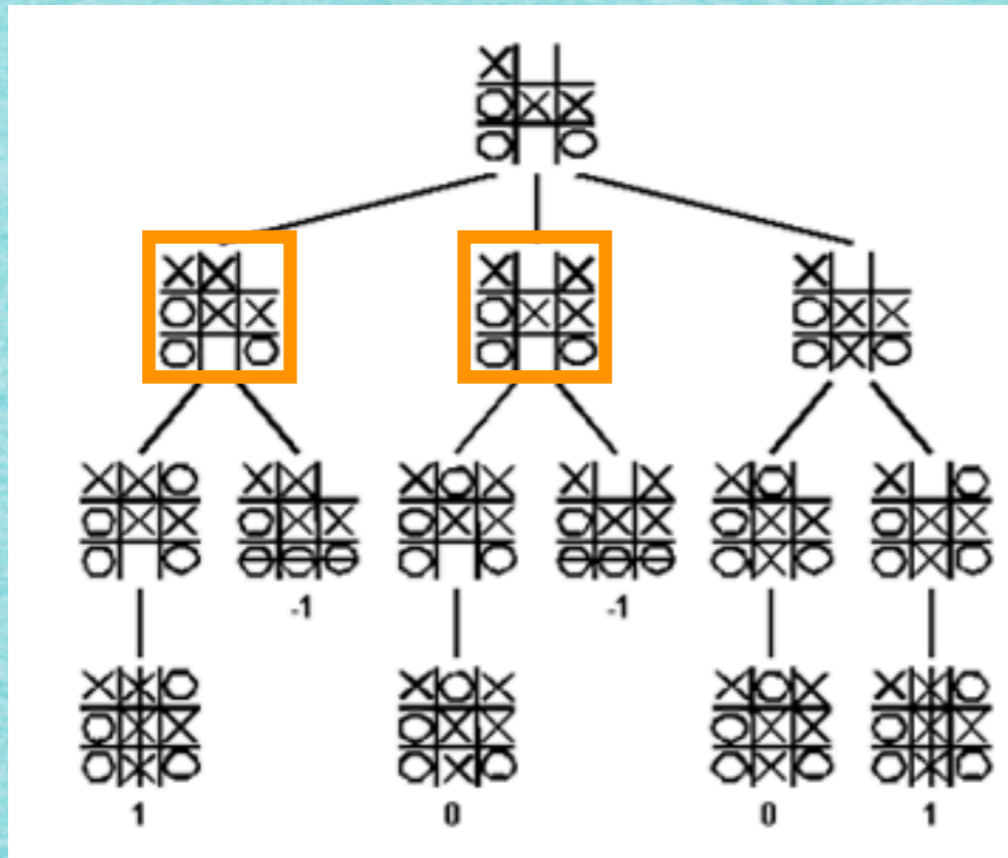
5.2 Schranken: Entscheidungsbäume



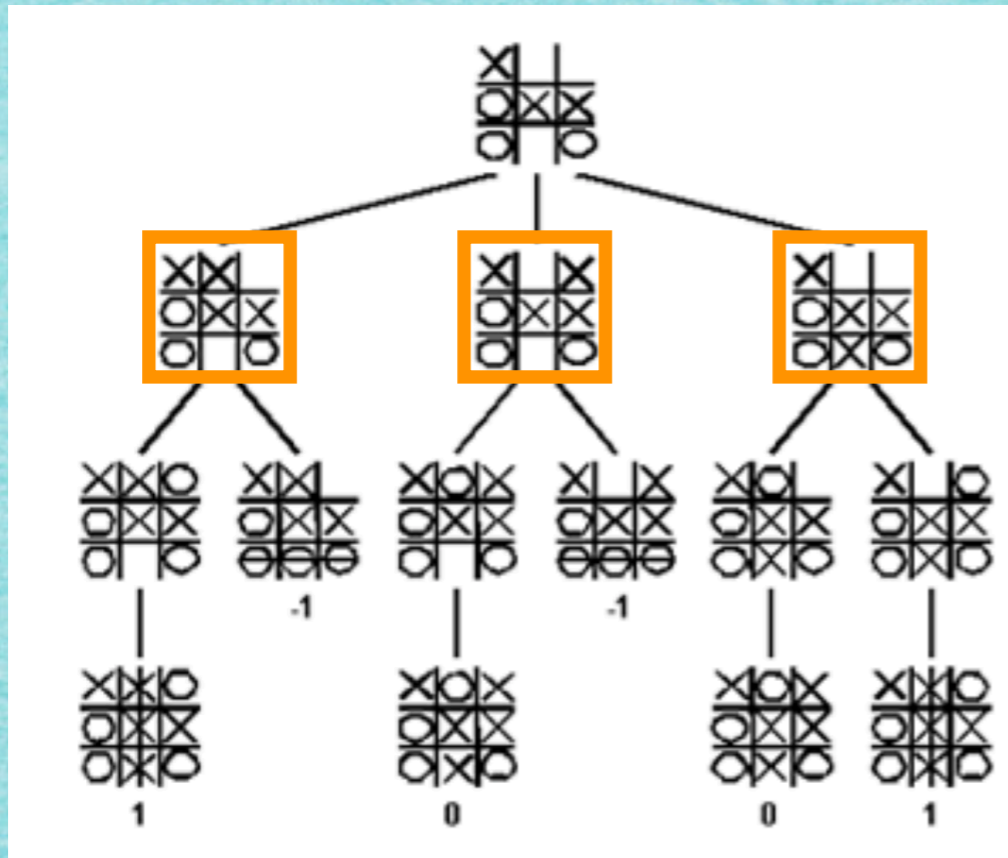
5.2 Schranken: Entscheidungsbäume



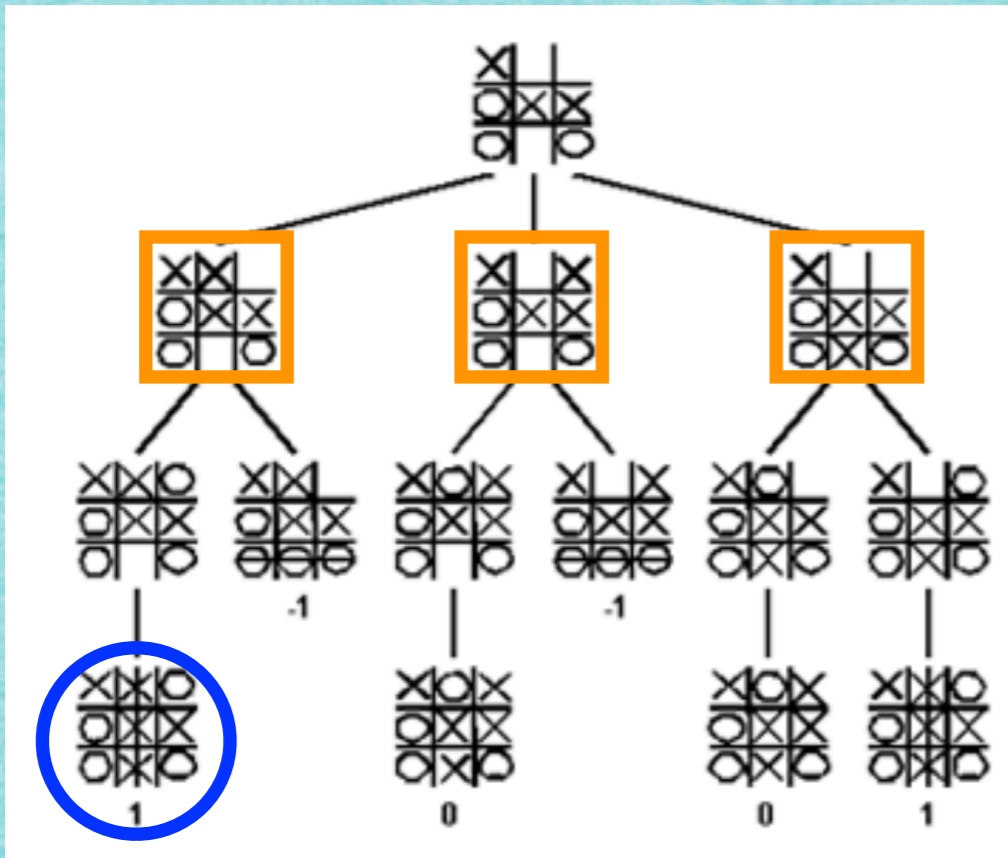
5.2 Schranken: Entscheidungsbäume



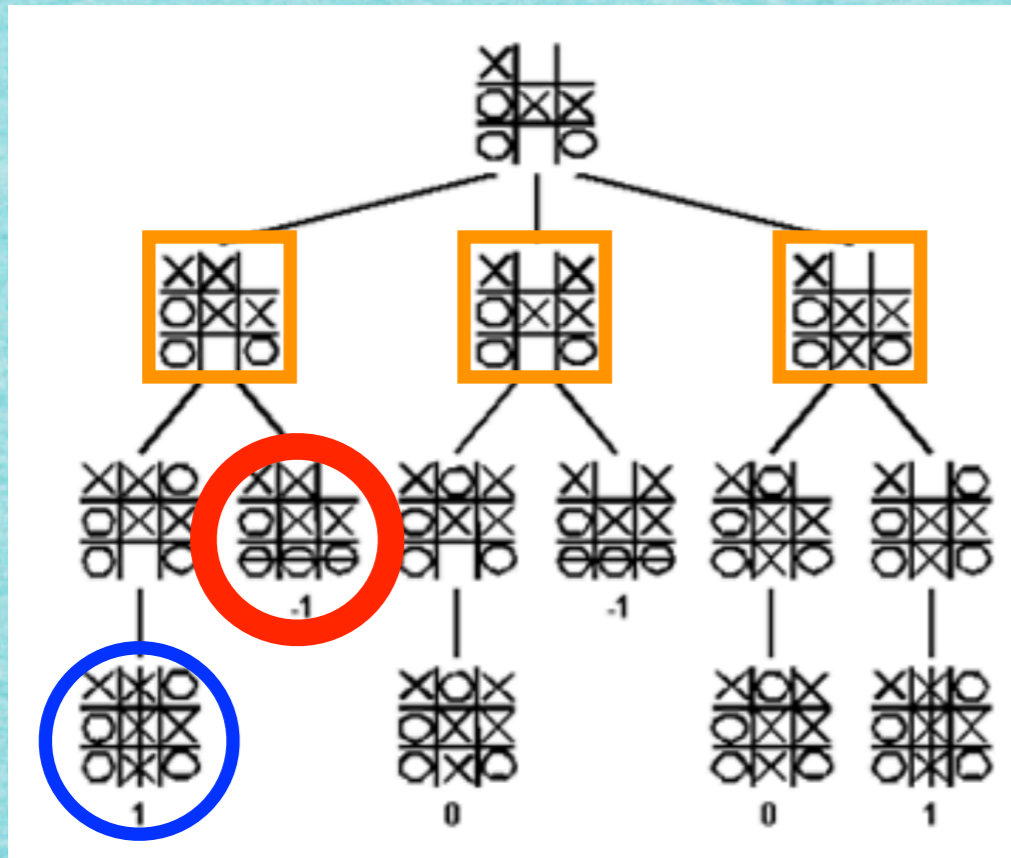
5.2 Schranken: Entscheidungsbäume



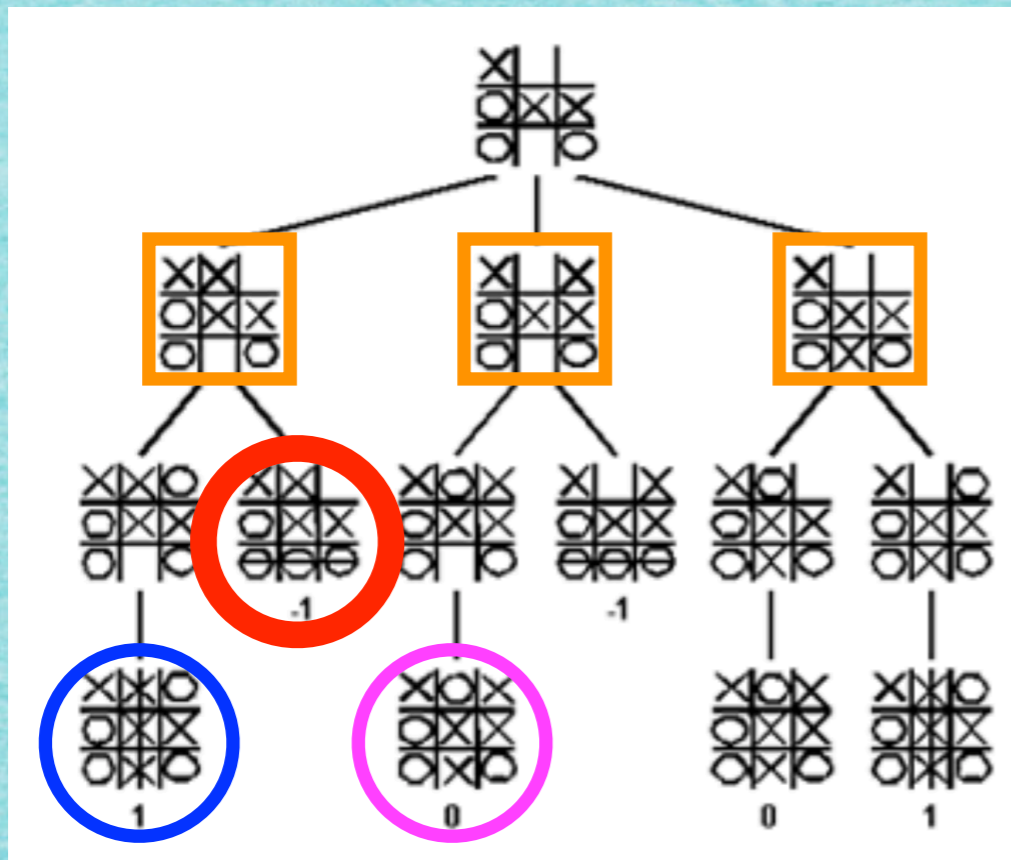
5.2 Schranken: Entscheidungsbäume



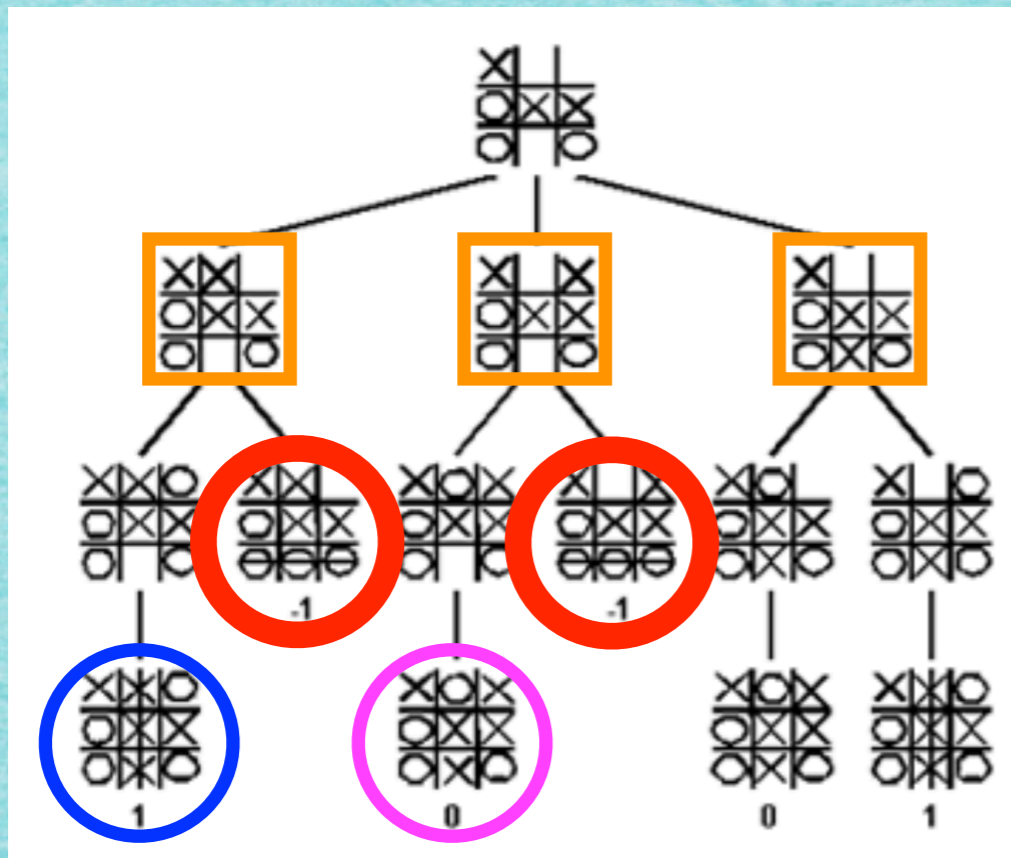
5.2 Schranken: Entscheidungsbäume



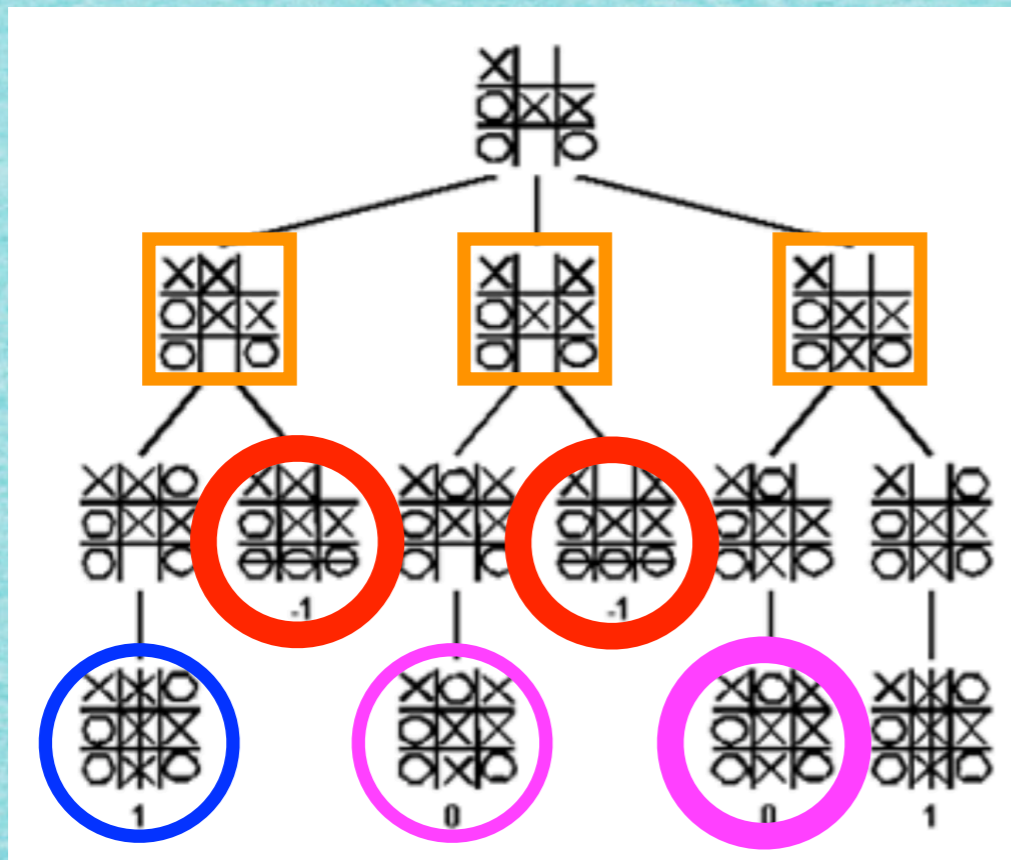
5.2 Schranken: Entscheidungsbäume



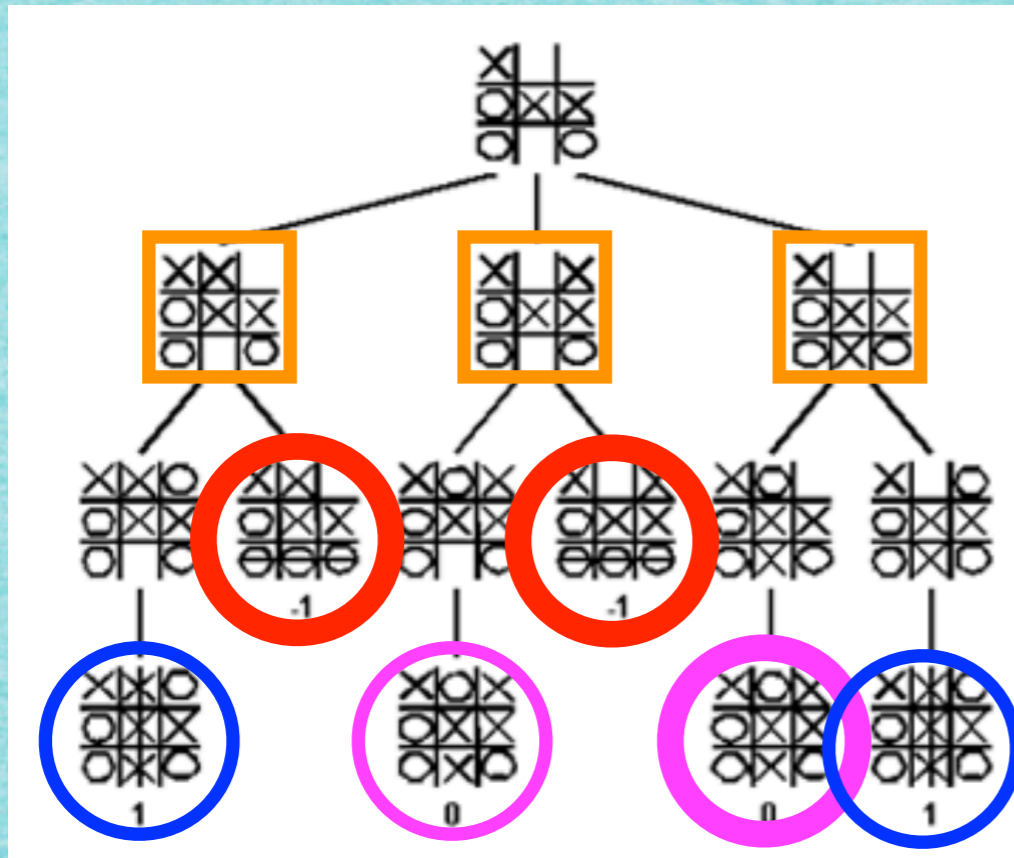
5.2 Schranken: Entscheidungsbäume



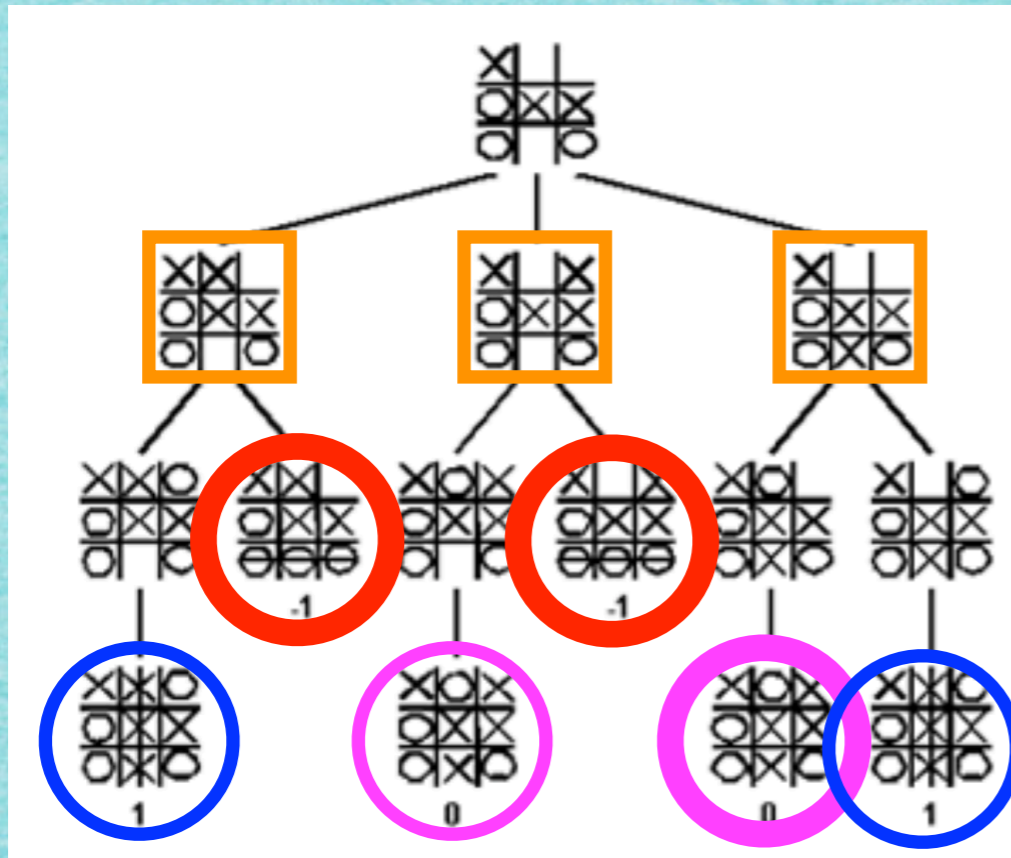
5.2 Schranken: Entscheidungsbäume



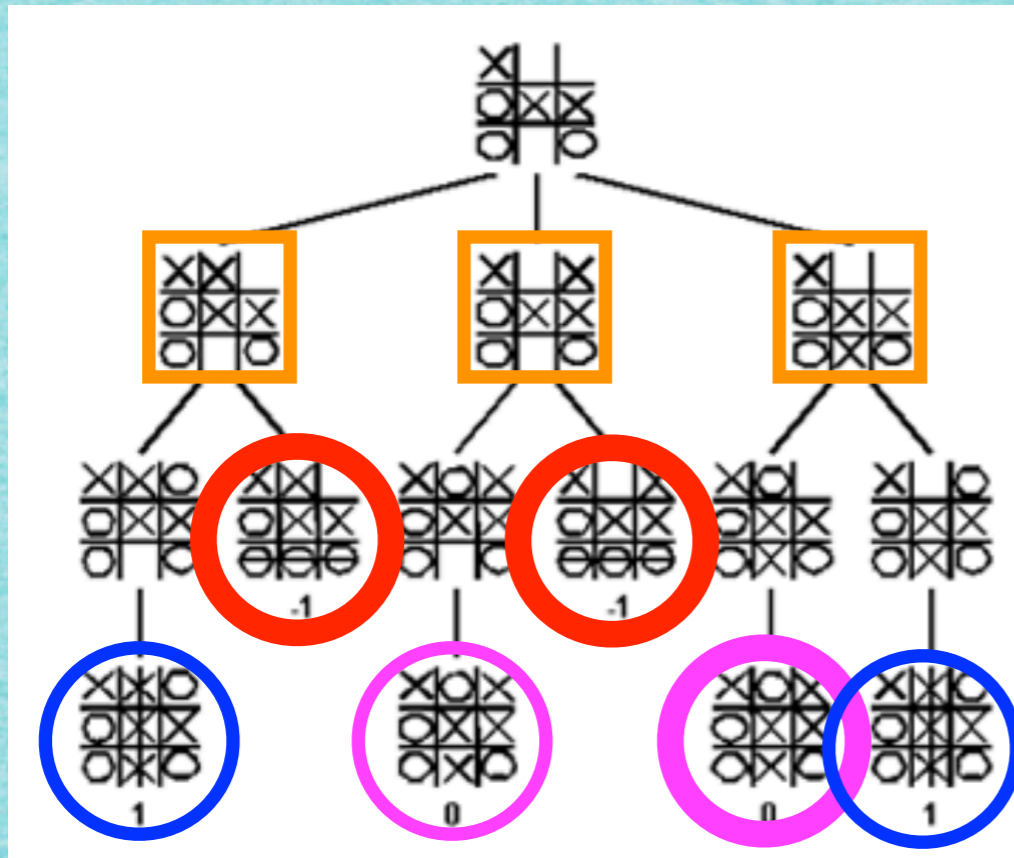
5.2 Schranken: Entscheidungsbäume



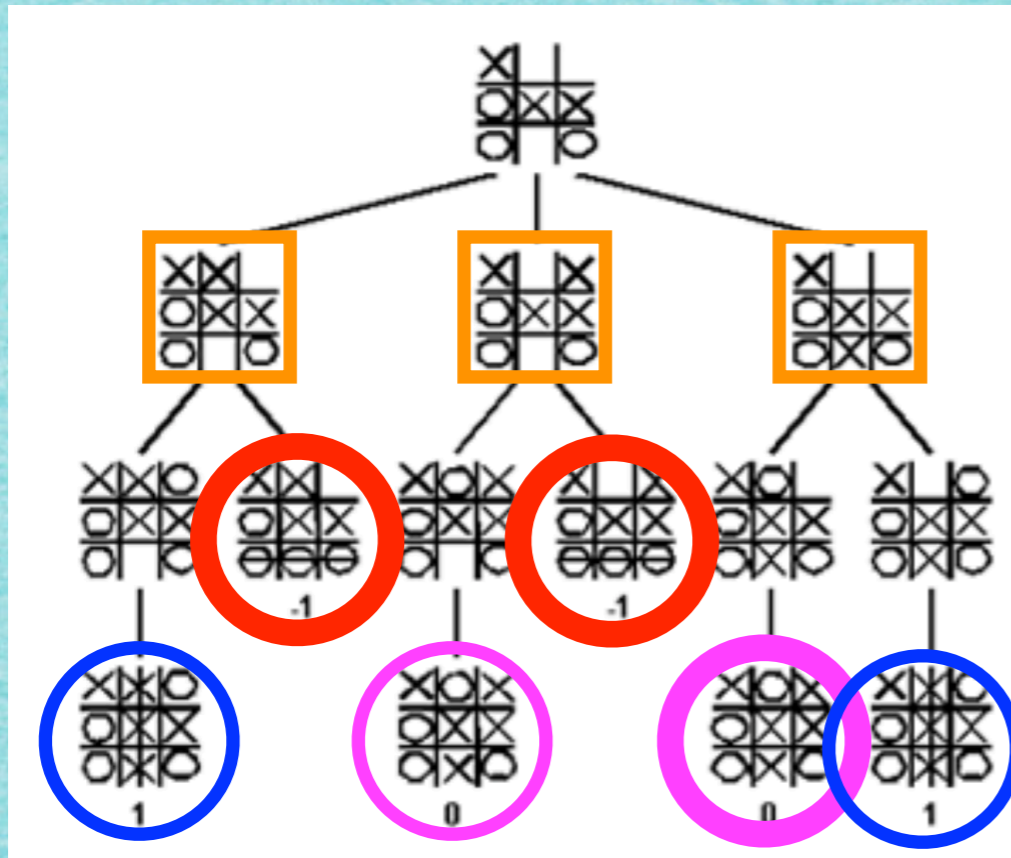
5.2 Schranken: Entscheidungsbäume



5.2 Schranken: Entscheidungsbäume



5.2 Schranken: Entscheidungsbäume



5.2 Schranken: Entscheidungsbäume



5.2 Schranken: Entscheidungsbäume



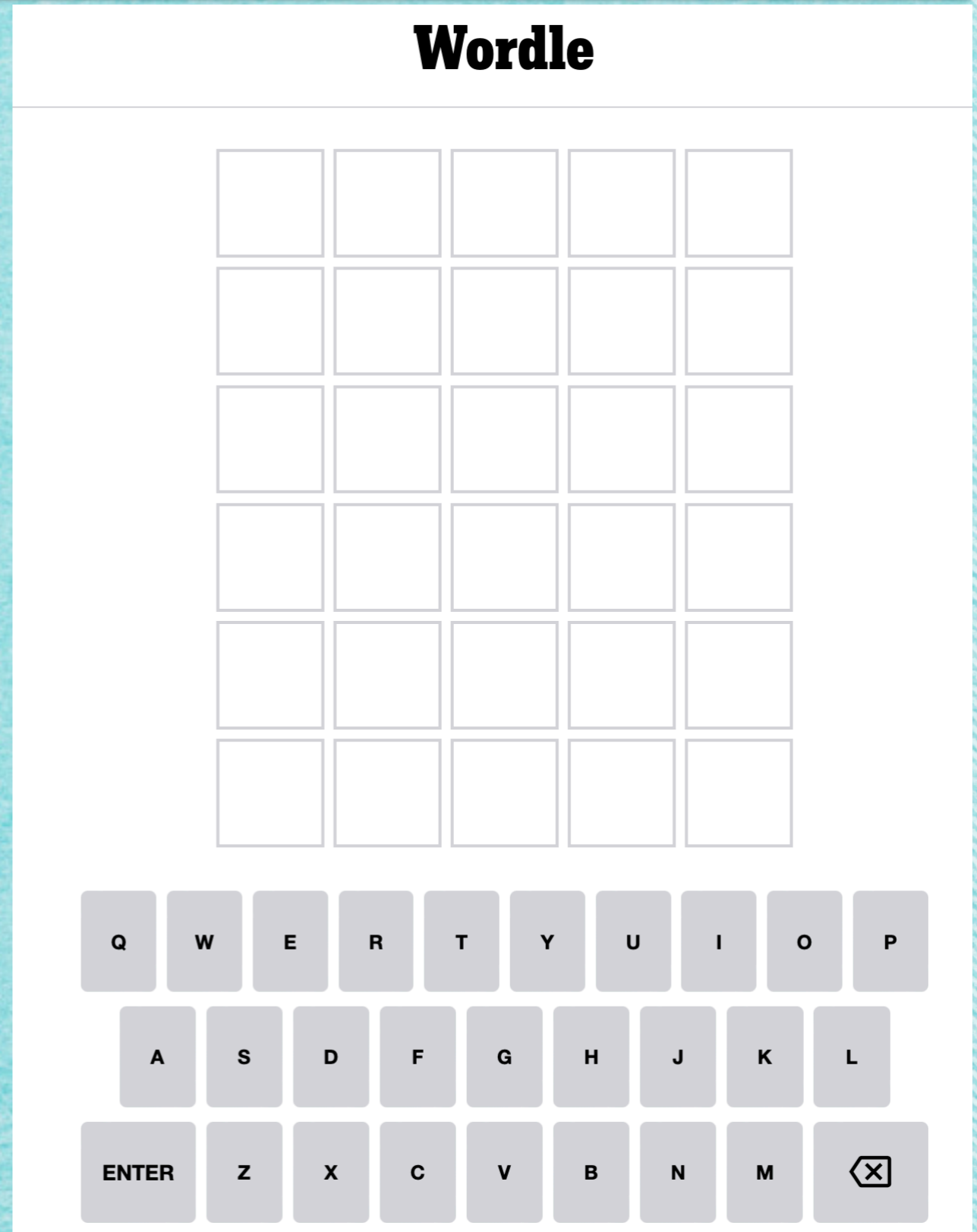
Wordle



5.2 Schranken: Entscheidungsbäume



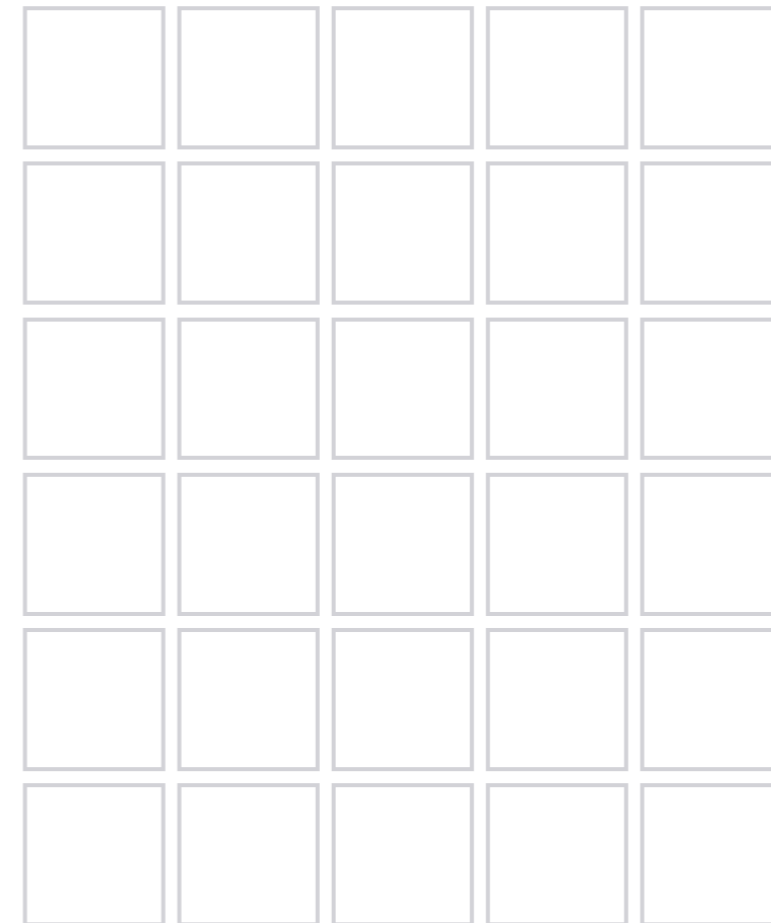
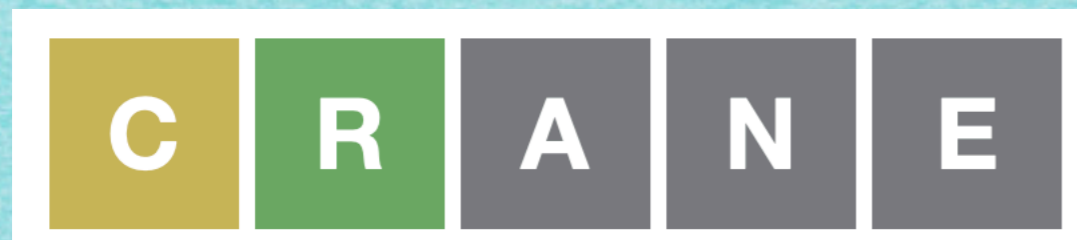
16.01.23



5.2 Schranken: Entscheidungsbäume



Wordle



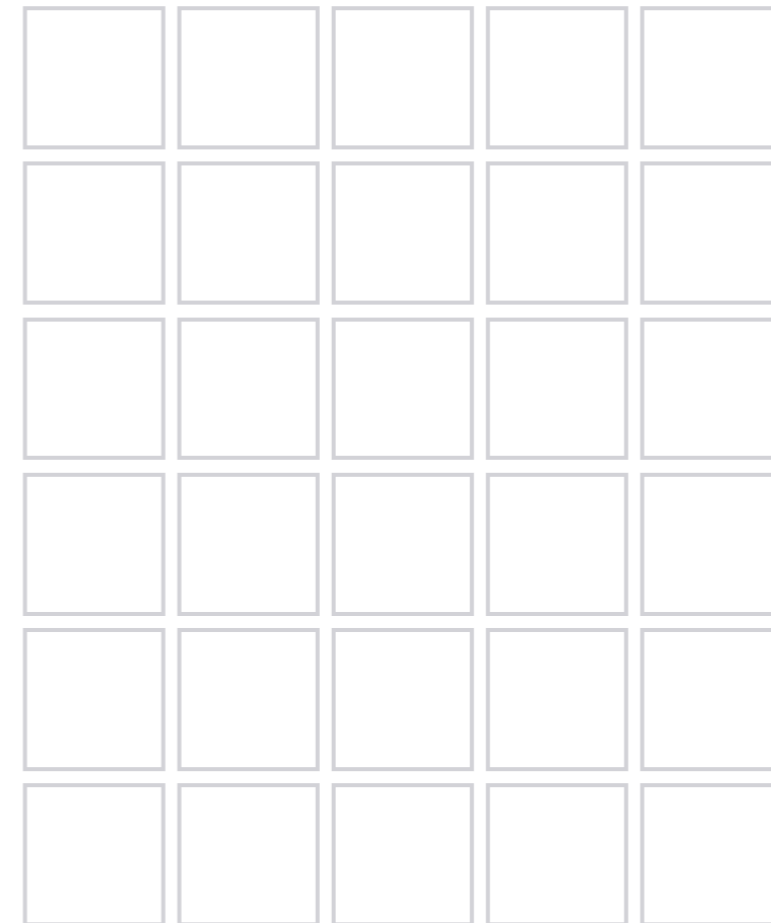
16.01.23



5.2 Schranken: Entscheidungsbäume



Wordle



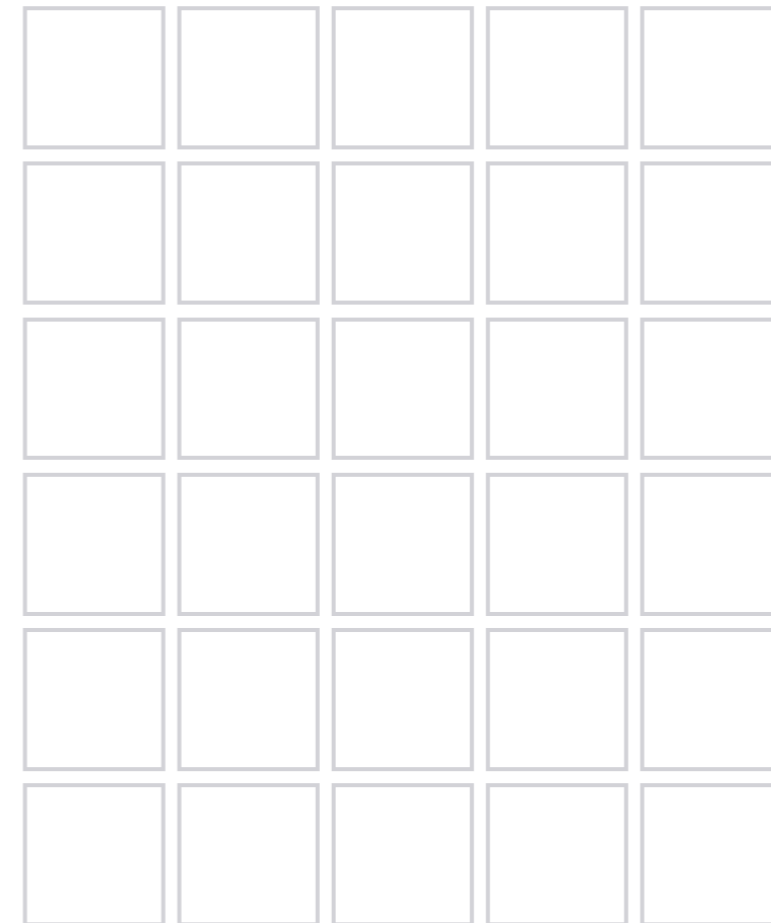
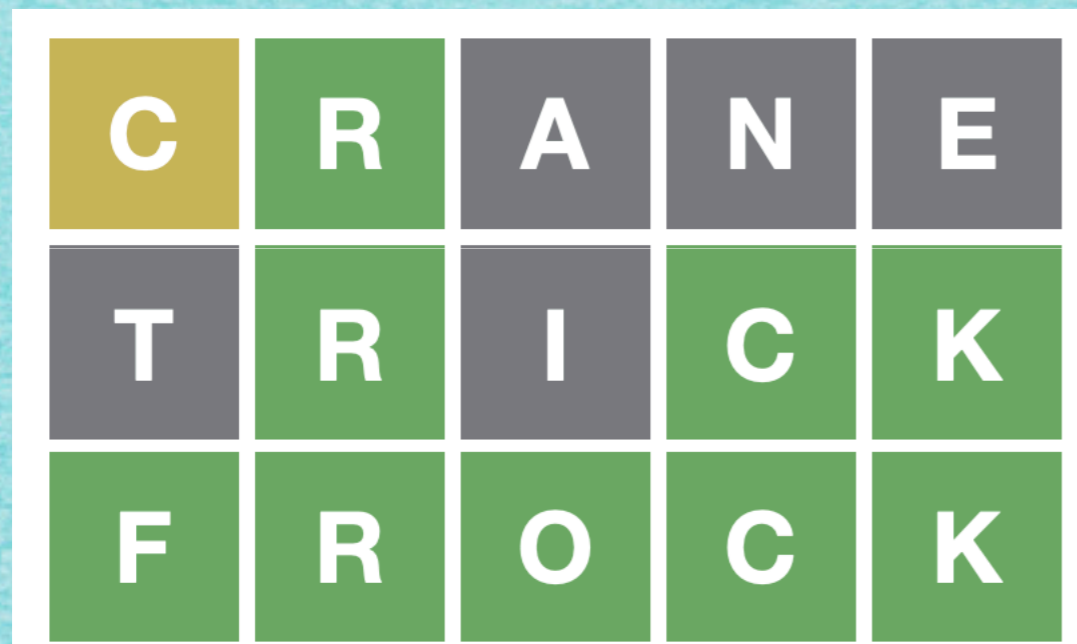
16.01.23



5.2 Schranken: Entscheidungsbäume



Wordle



16.01.23



5.2 Schranken: Entscheidungsbäume



C	R	A	N	E
T	R	I	C	K
F	R	O	C	K

16.01.23

5.2 Schranken: Entscheidungsbäume

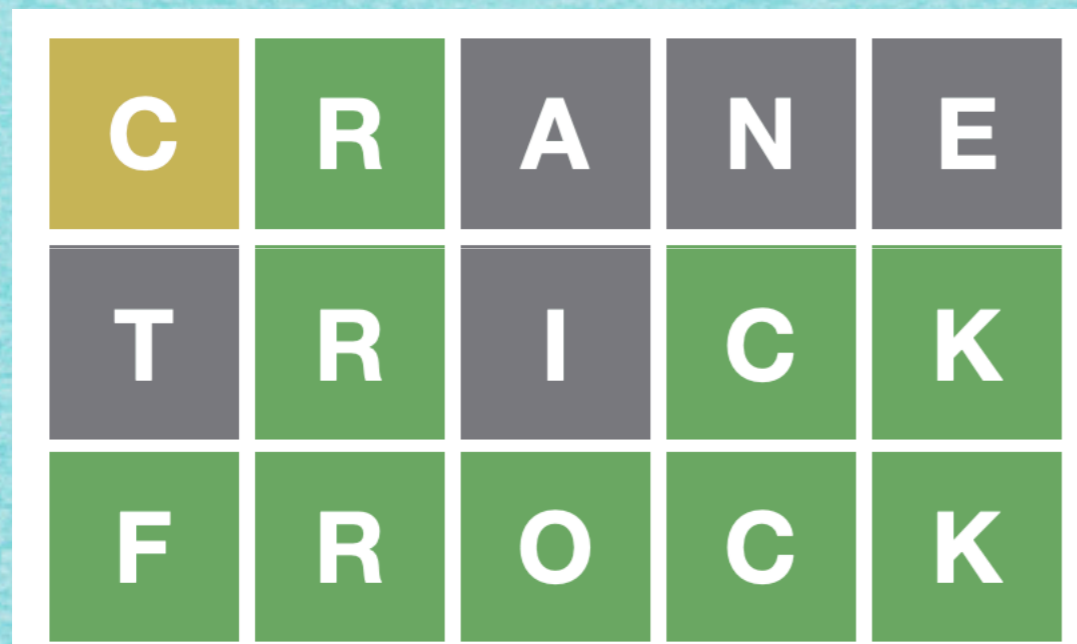


C	R	A	N	E
T	R	I	C	K
F	R	O	C	K

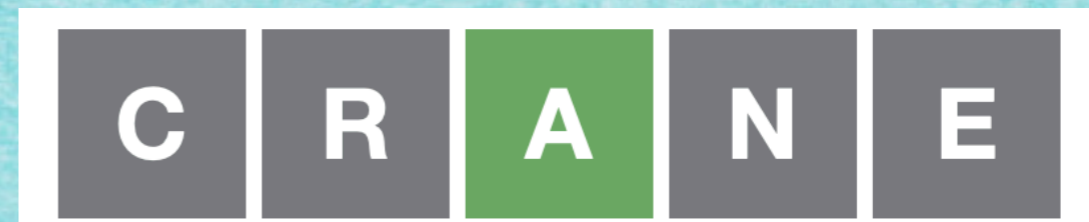
16.01.23

14.01.23

5.2 Schranken: Entscheidungsbäume

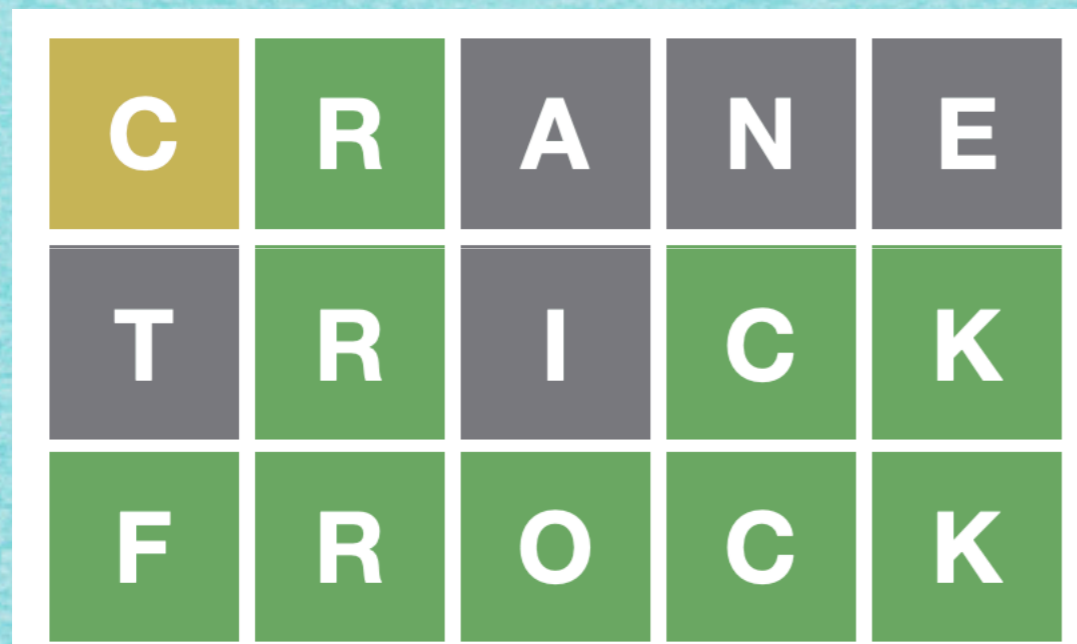


16.01.23



14.01.23

5.2 Schranken: Entscheidungsbäume

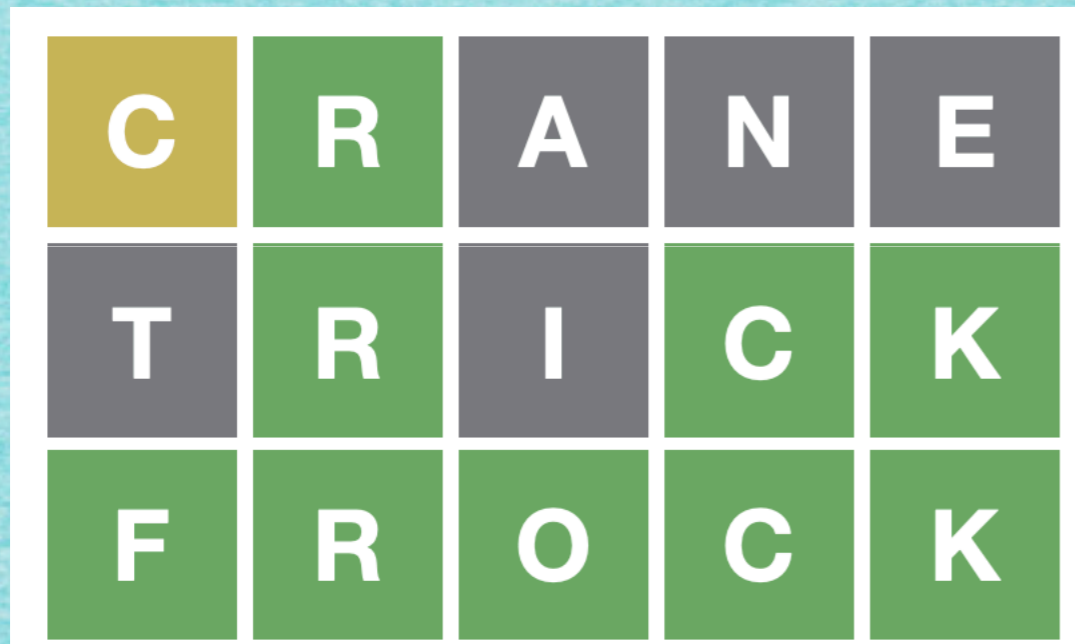


16.01.23



14.01.23

5.2 Schranken: Entscheidungsbäume



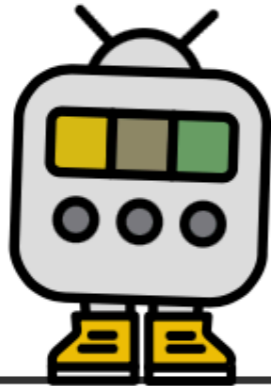
16.01.23



14.01.23

WordleBot: Improve Your Wordle Strategy

By Josh Katz and Matthew Conlen



What would you like to do?

17.01.23



Wordle



17.01.23

Q	W	E	R	T	Y	U	I	O	P
A	S	D	F	G	H	J	K	L	
ENTER	Z	X	C	V	B	N	M	⌫	



Weekend Diversion: The Logic That Stumped Brooklyn Nine Nine



Ethan Siegel [Follow](#)

May 24, 2015 · 11 min read ★



The most famous logic puzzle from the best police comedy on television, and how to (finally) solve it!





“There are 12 men on an island. 11 weigh exactly the same amount, but one of them is slightly lighter or heavier. You must figure out which. The island has no escapes, but there is a see-saw. The exciting catch? You can only use it three times.”

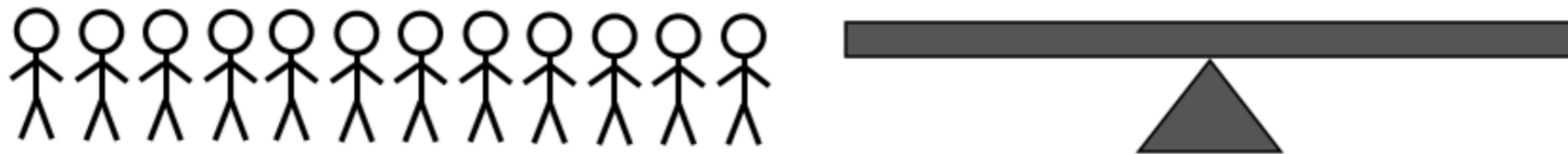
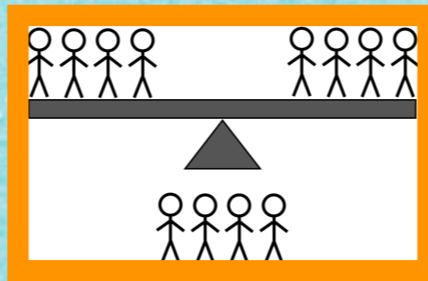
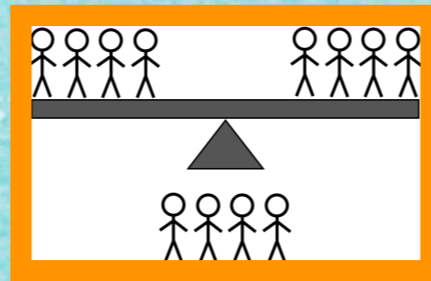


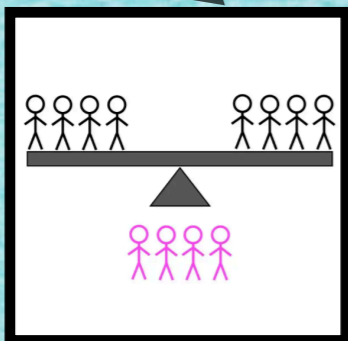
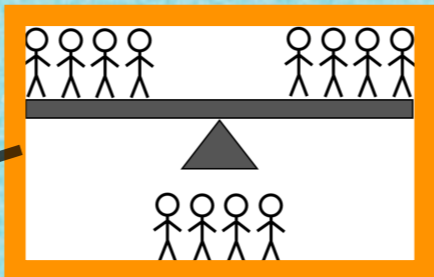
Image credit: E. Siegel, of the setup and starting materials.



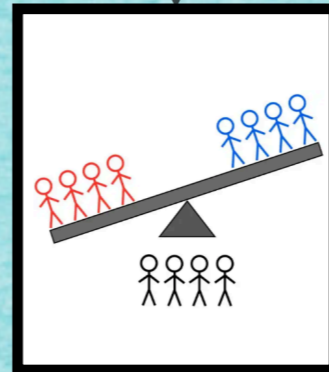
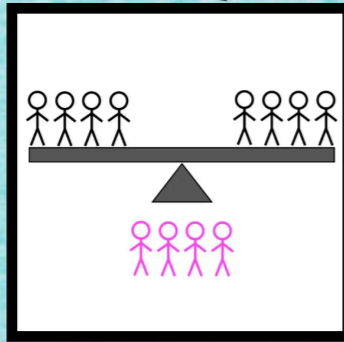
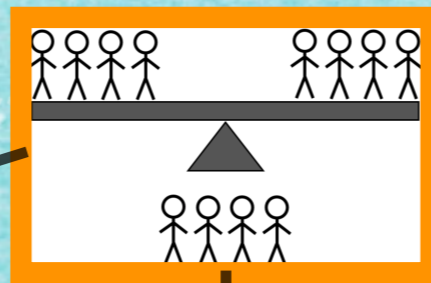
24



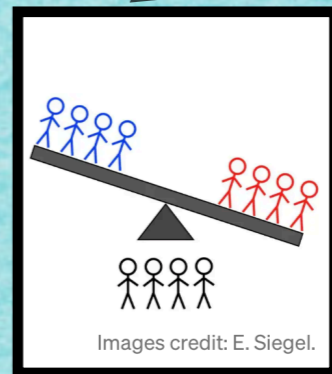
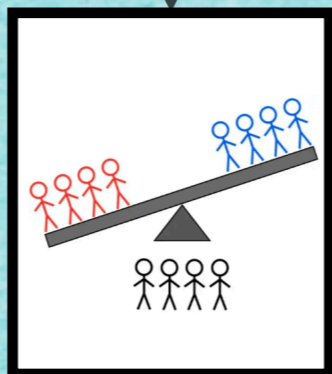
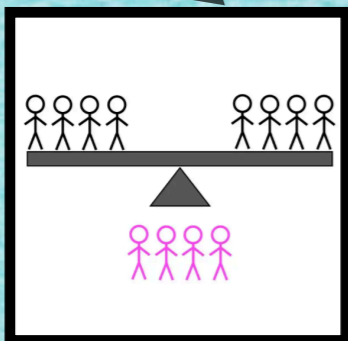
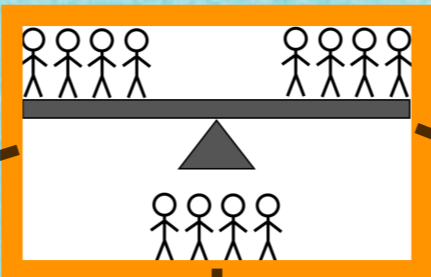
24



24



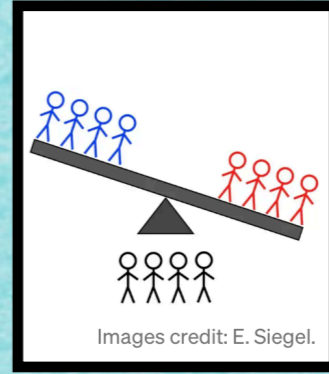
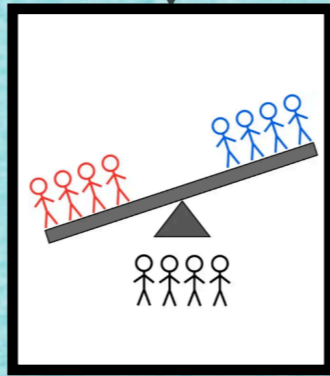
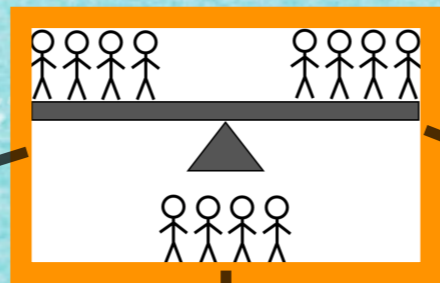
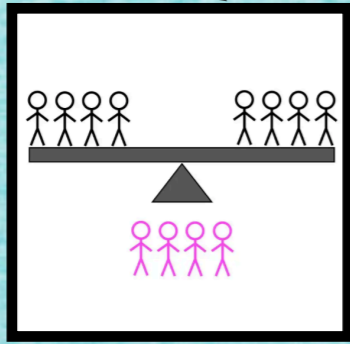
24



Images credit: E. Siegel.

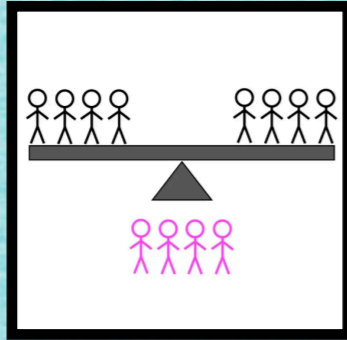
8

24

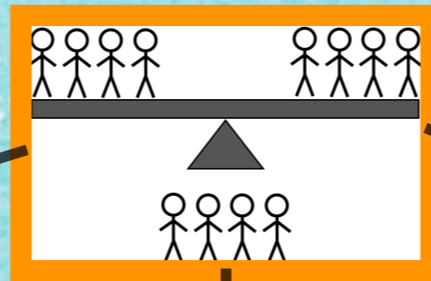


Images credit: E. Siegel.

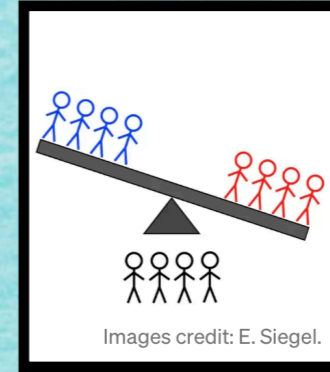
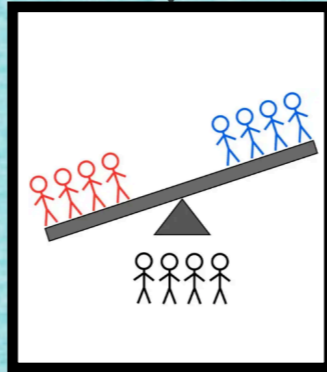
8



24

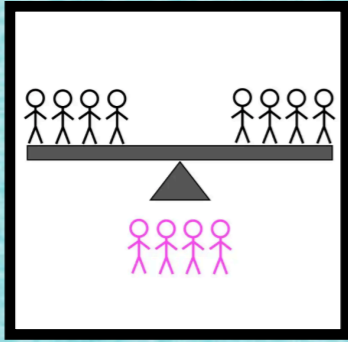


8

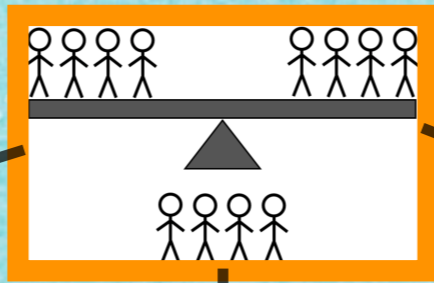


Images credit: E. Siegel.

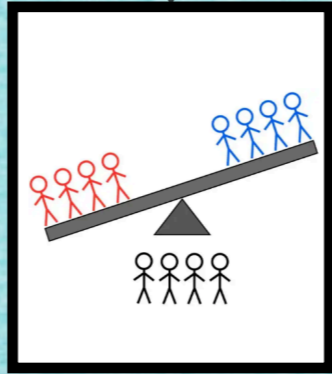
8



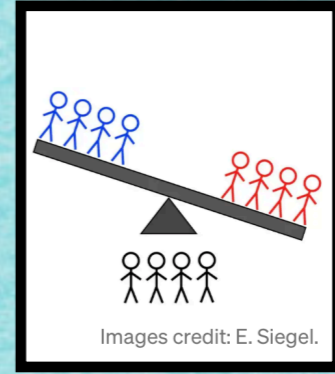
24



8

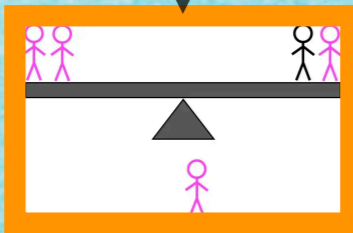
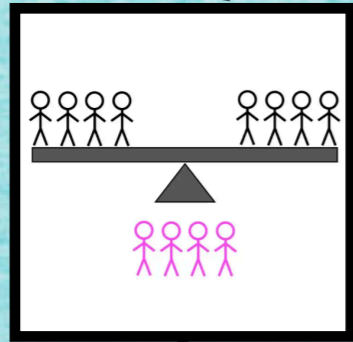


8

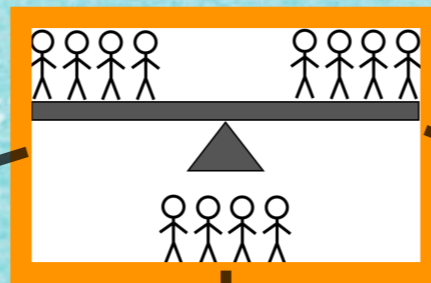


Images credit: E. Siegel.

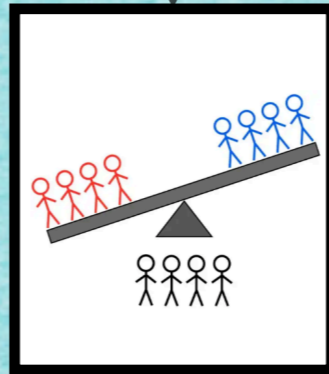
8



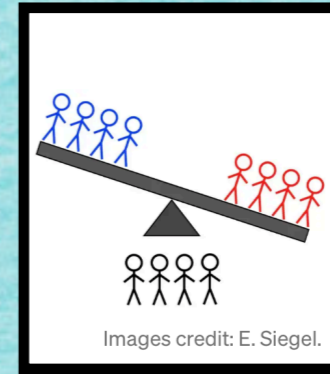
24



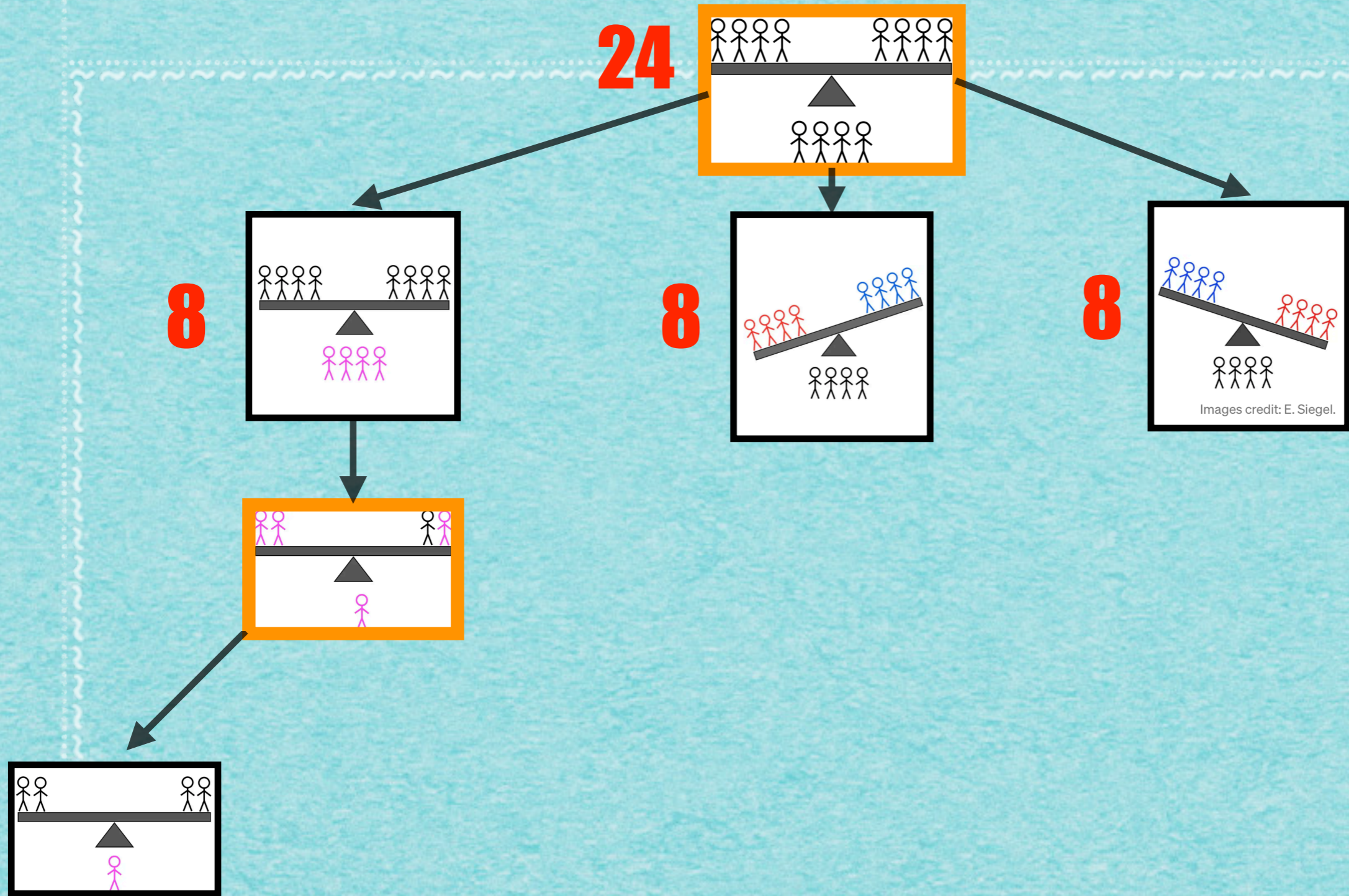
8



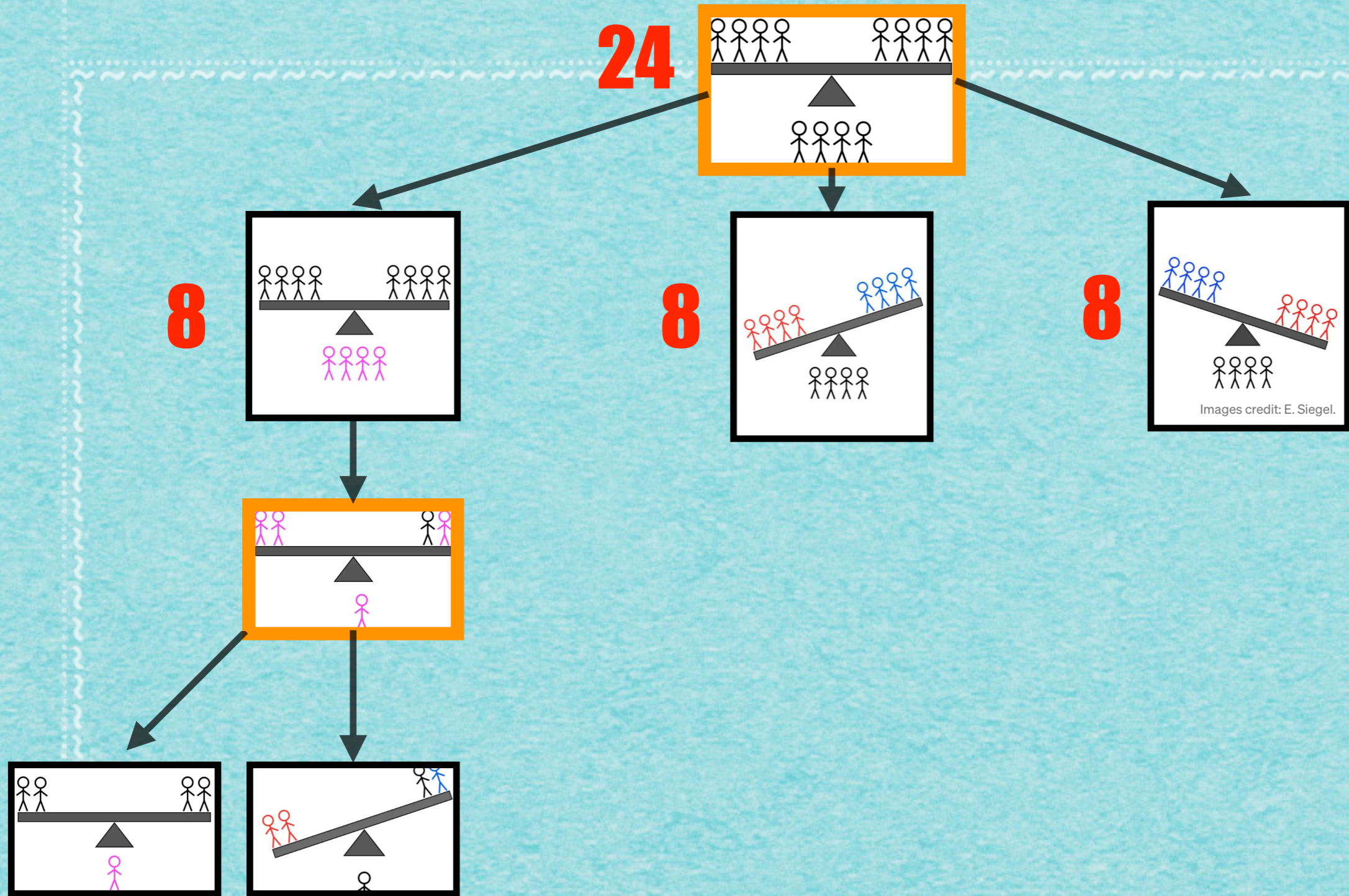
8

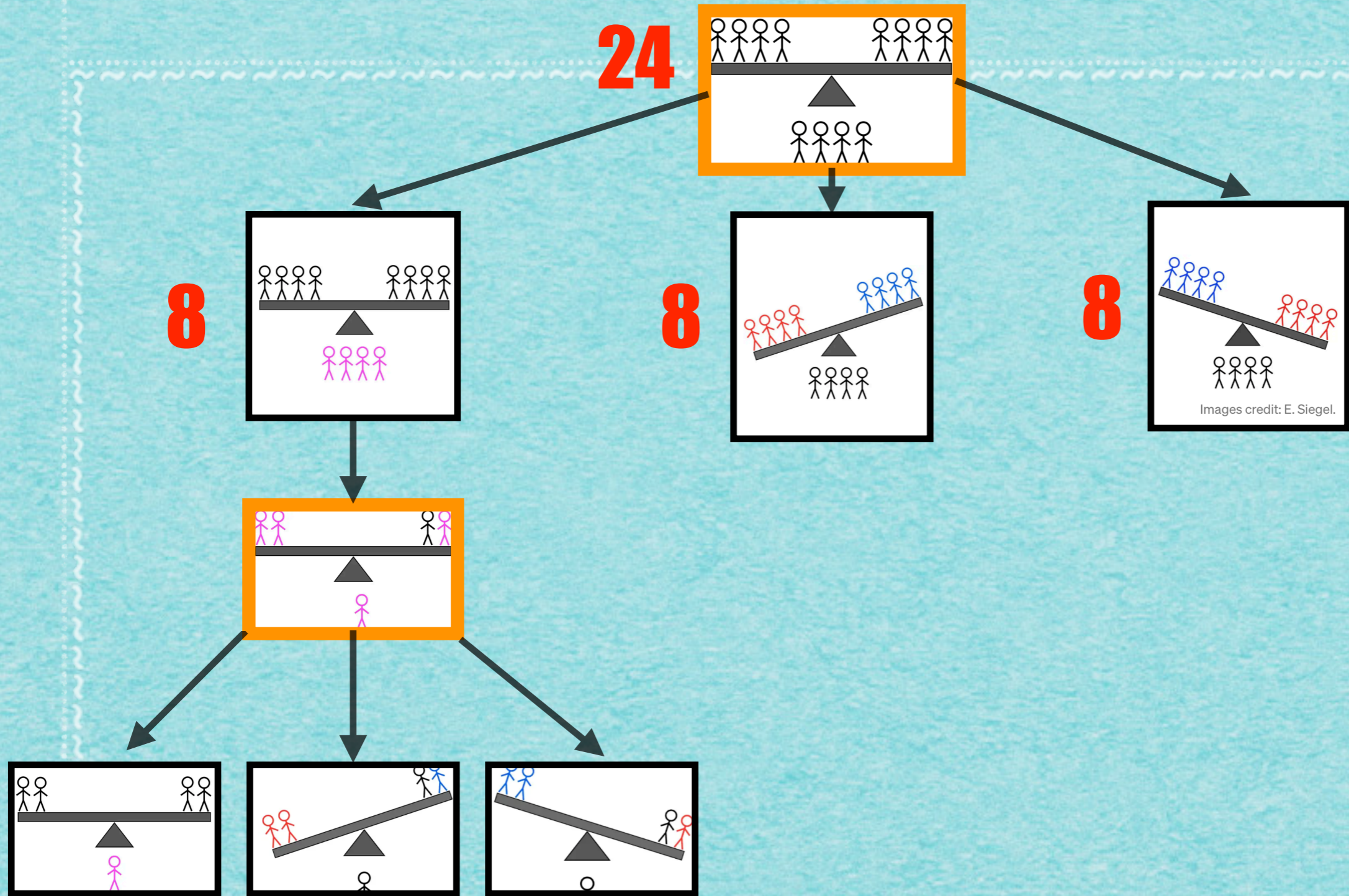


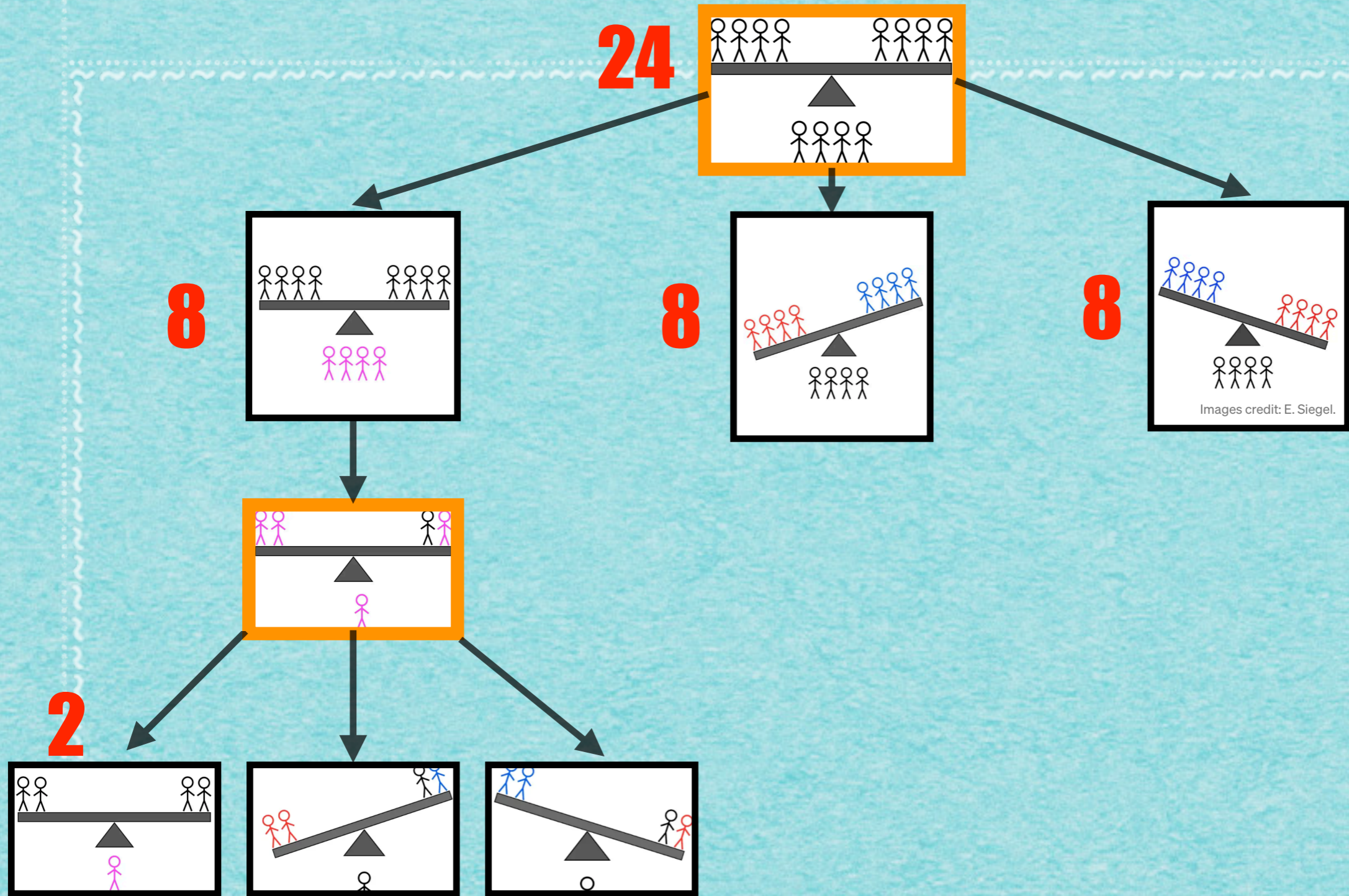
Images credit: E. Siegel.

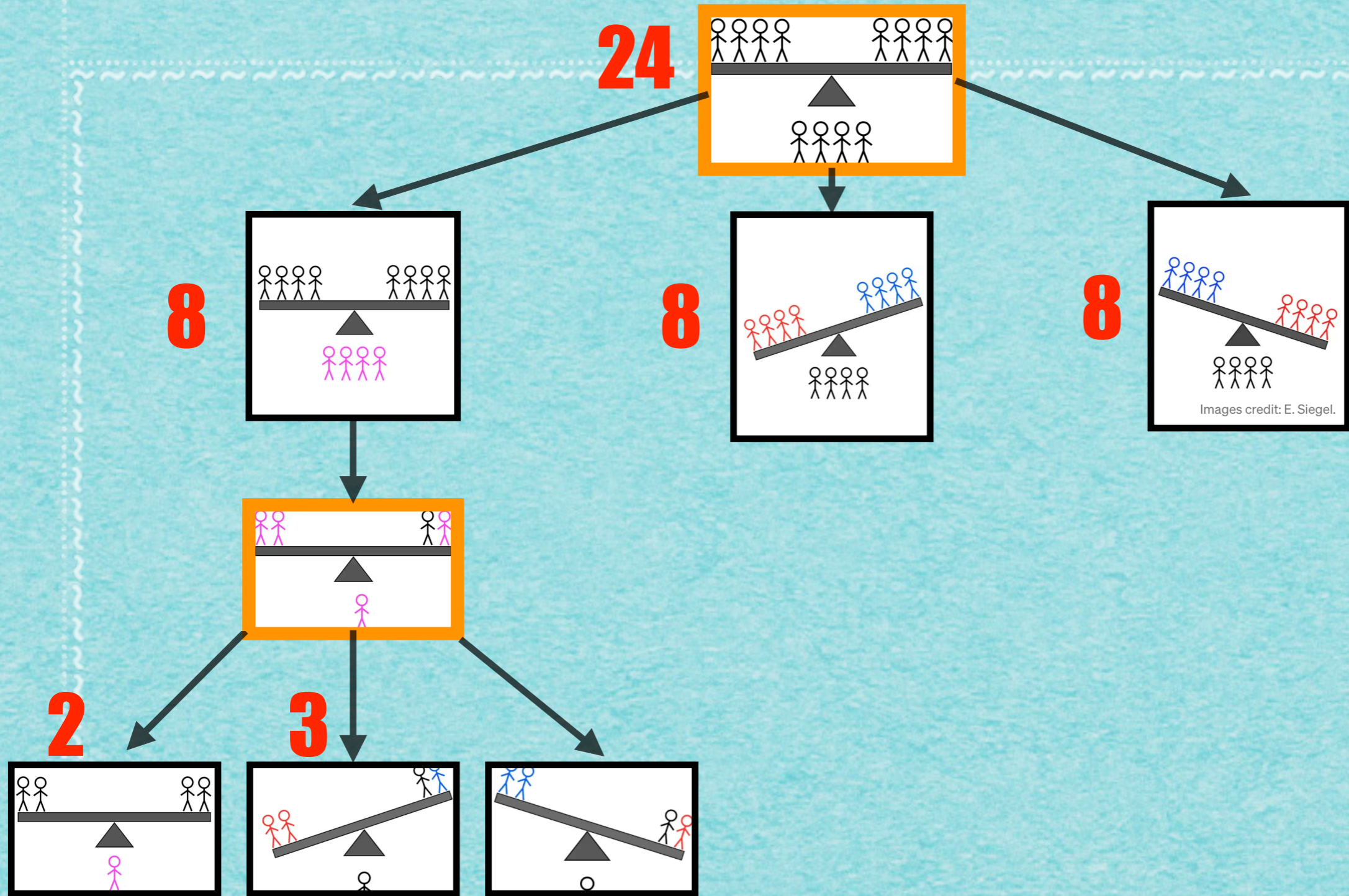


Images credit: E. Siegel.

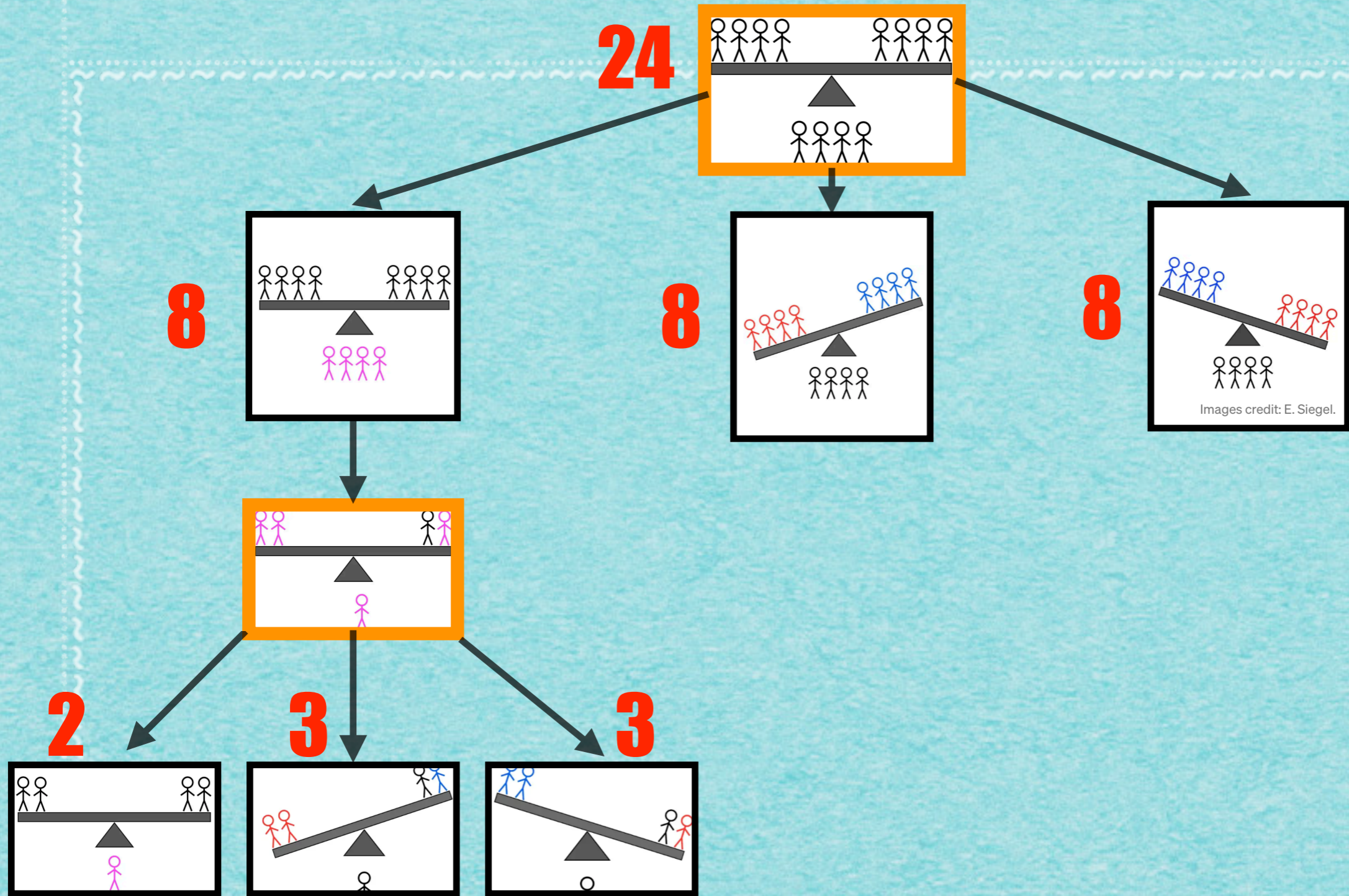




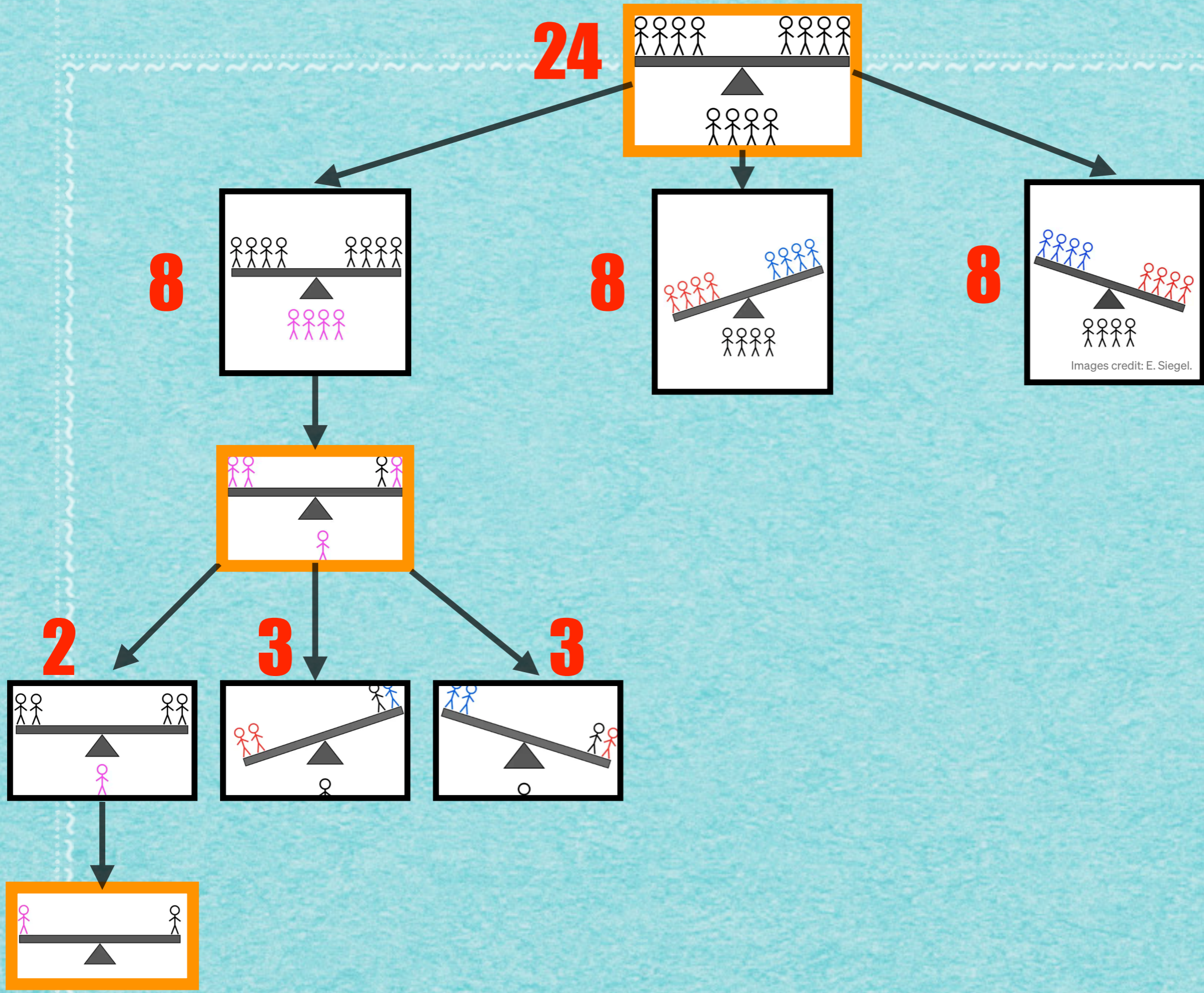


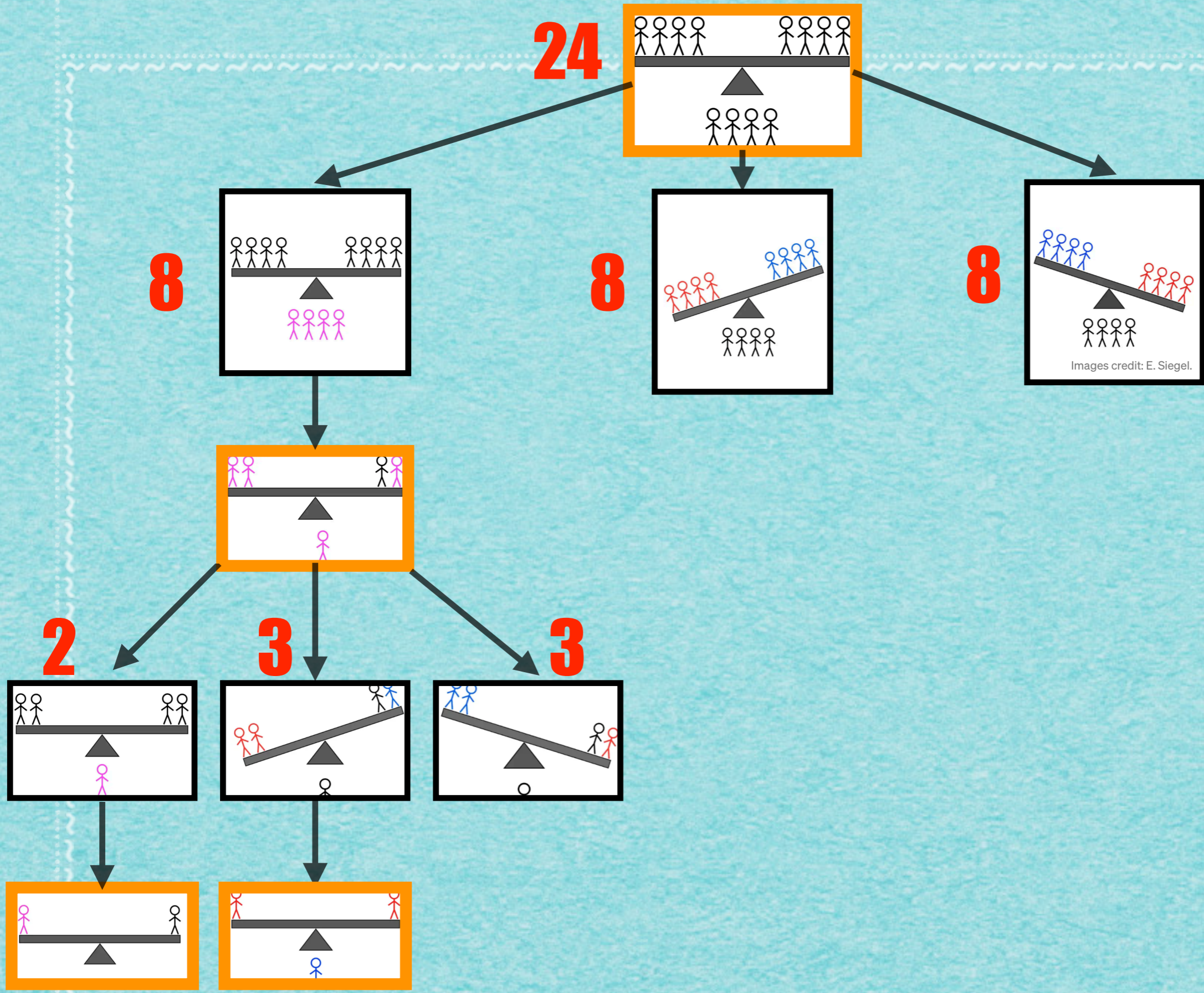


Images credit: E. Siegel.

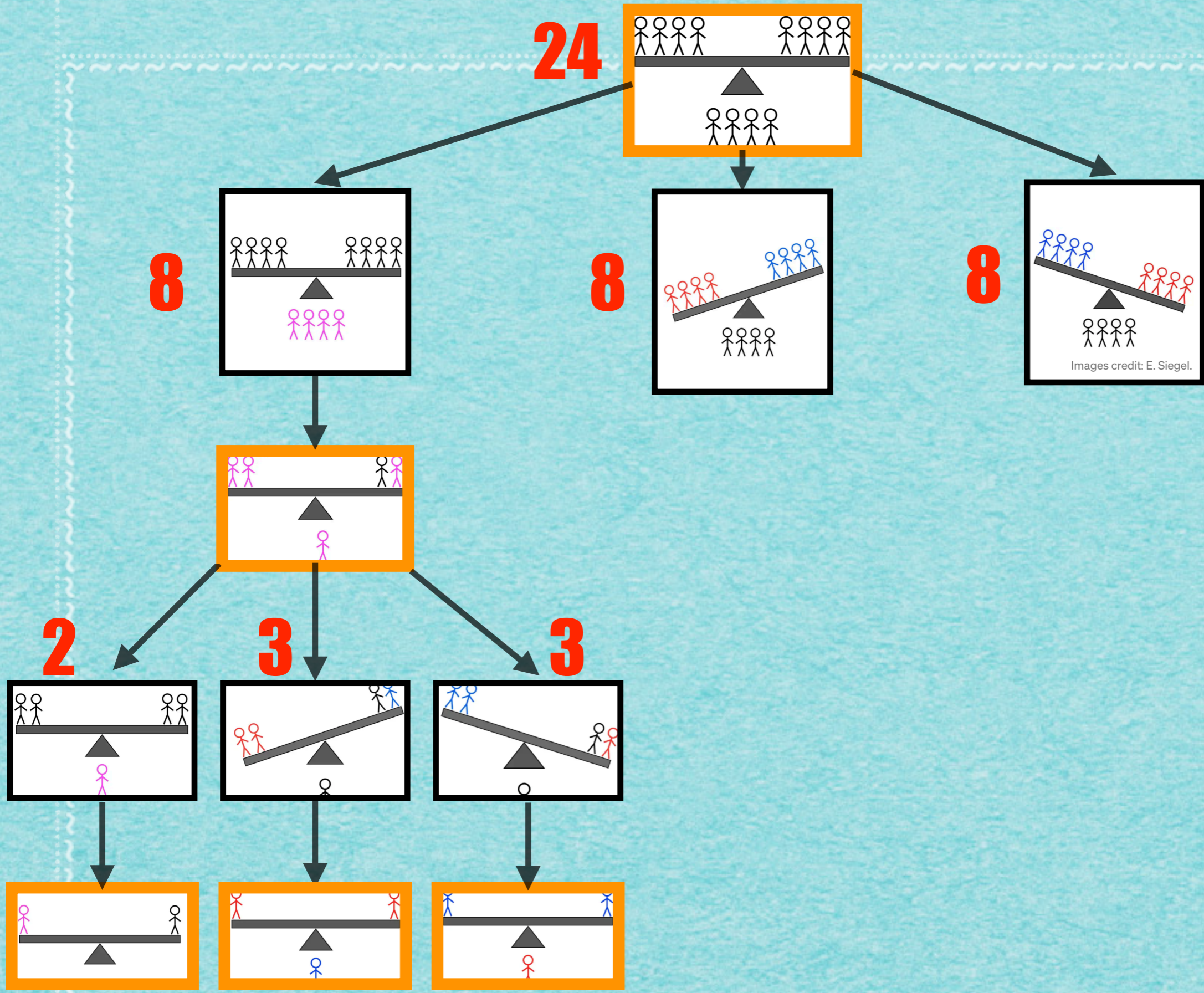


Images credit: E. Siegel.

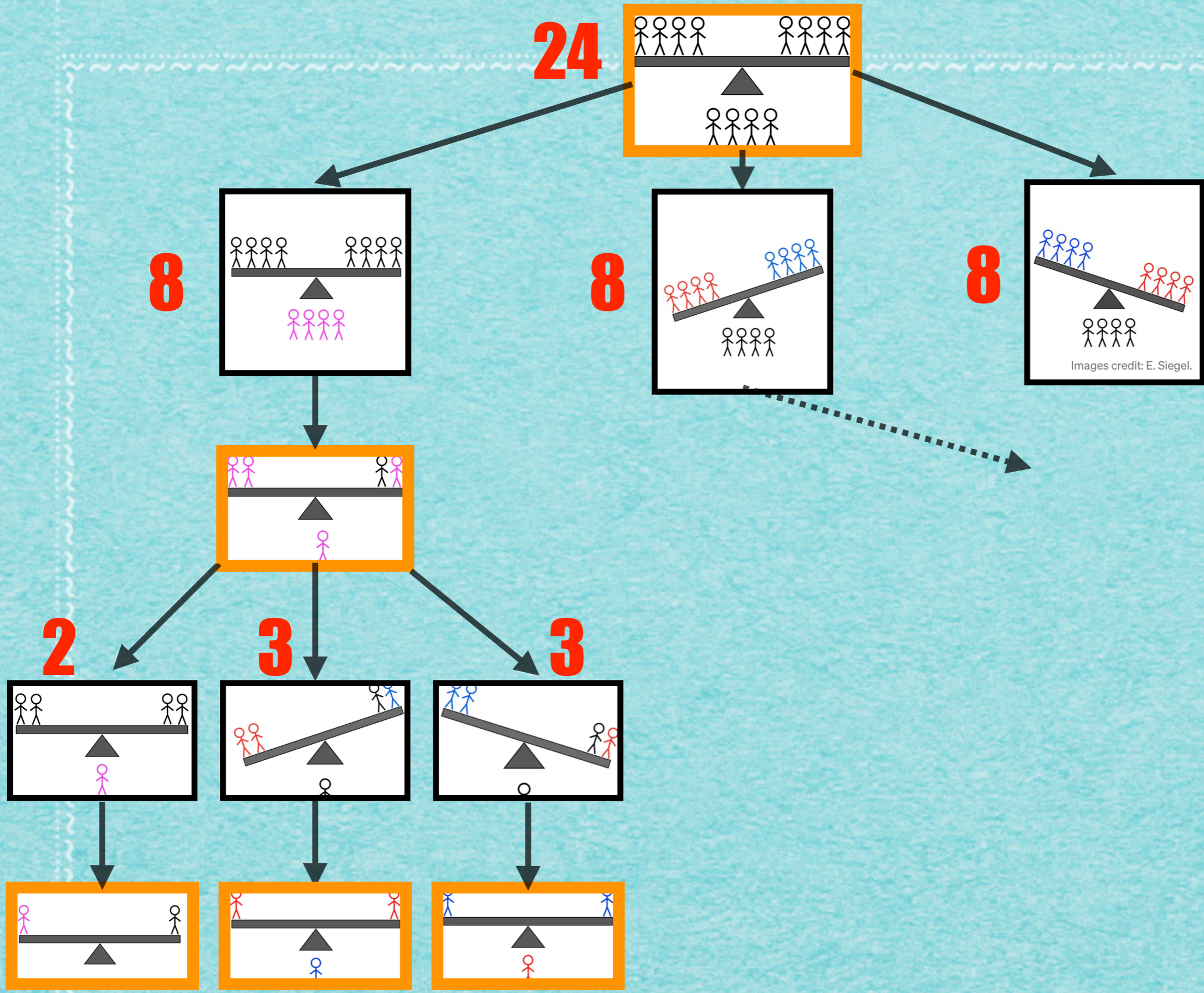




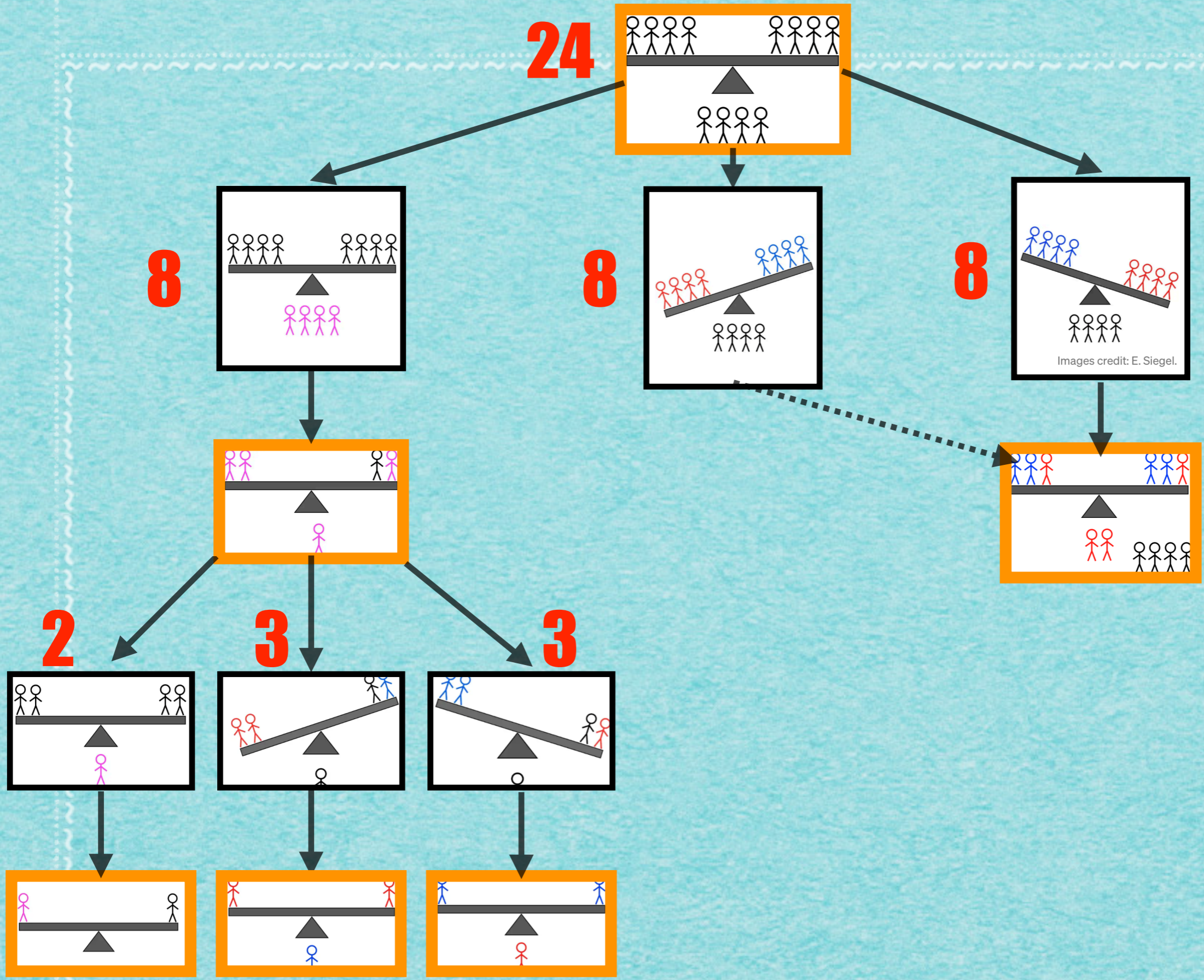
Images credit: E. Siegel.

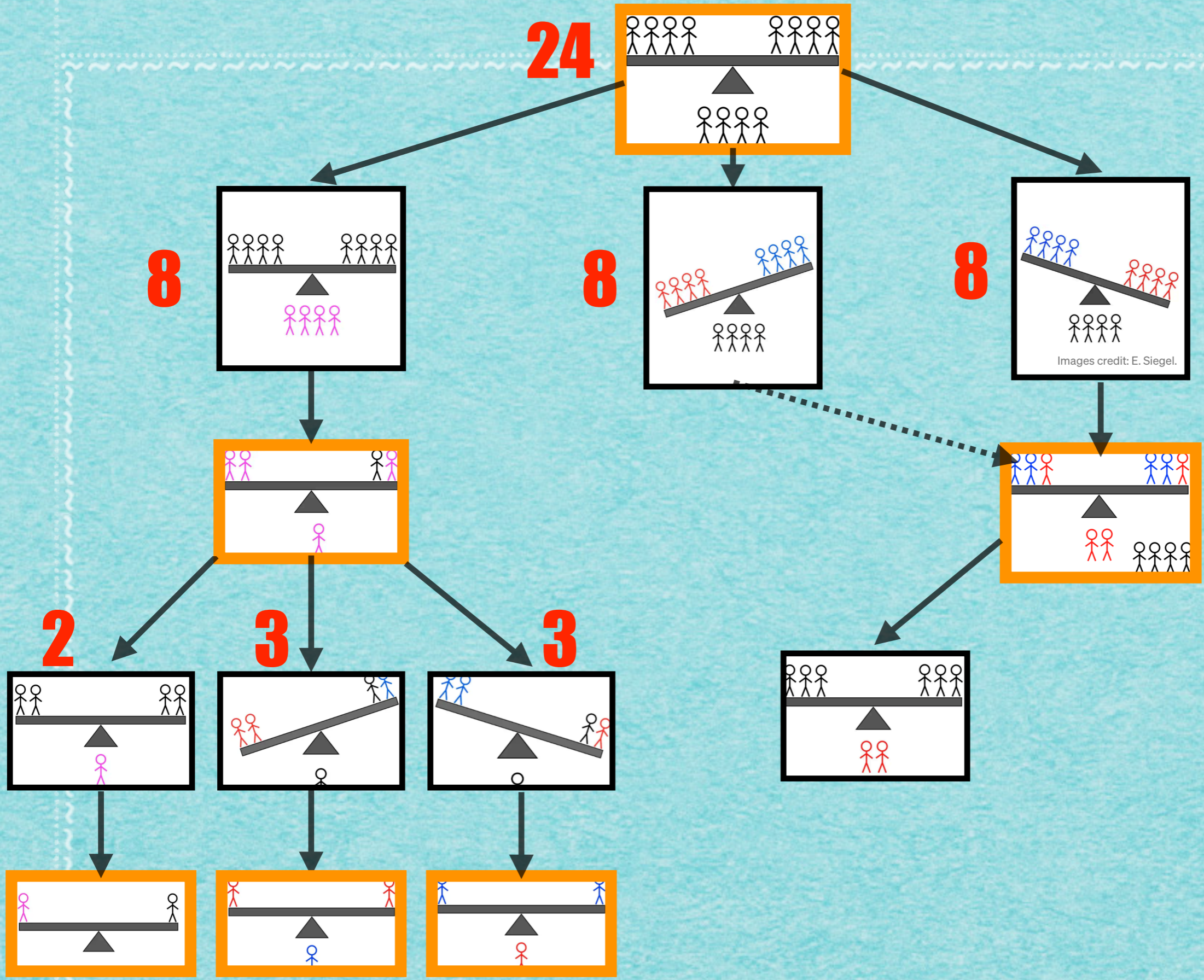


Images credit: E. Siegel.

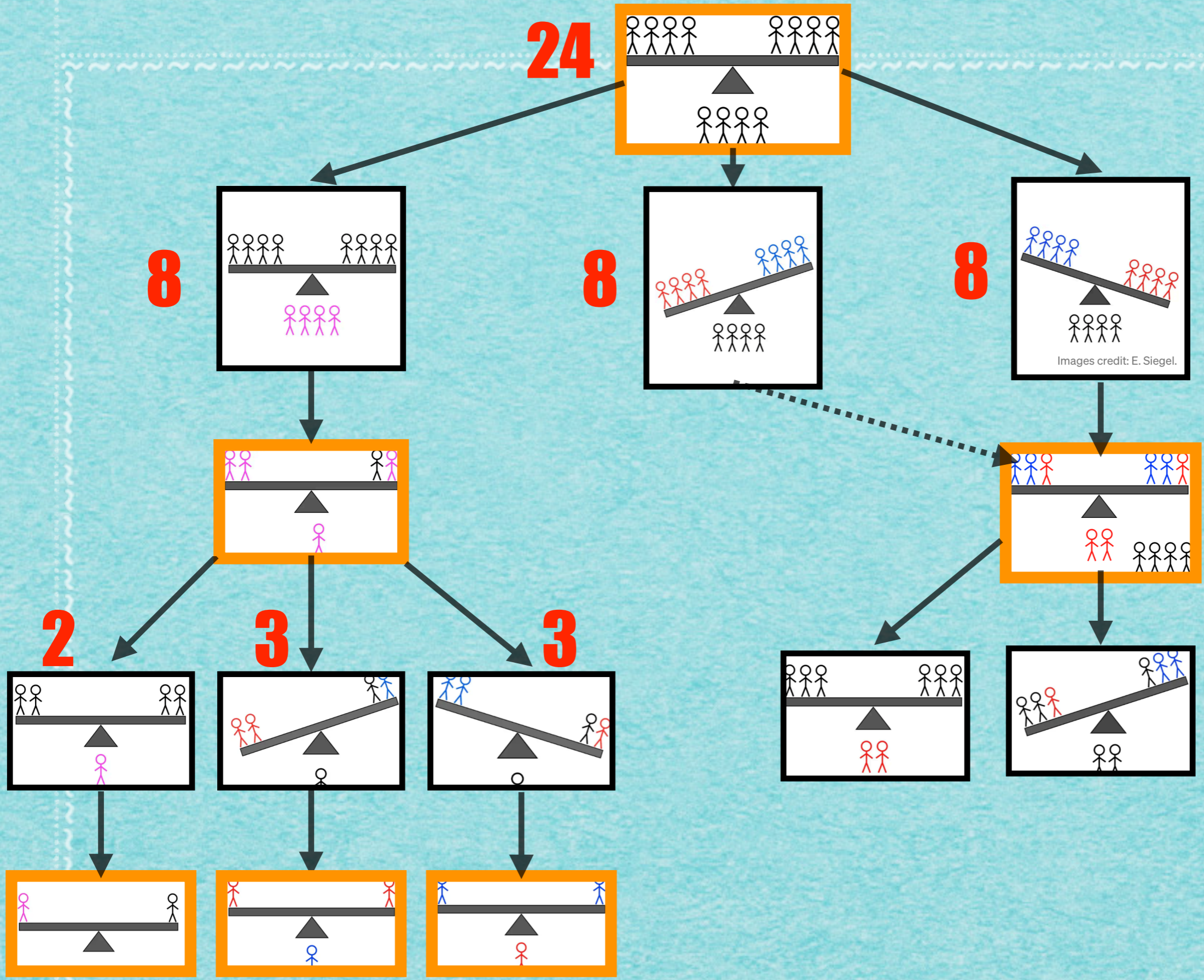


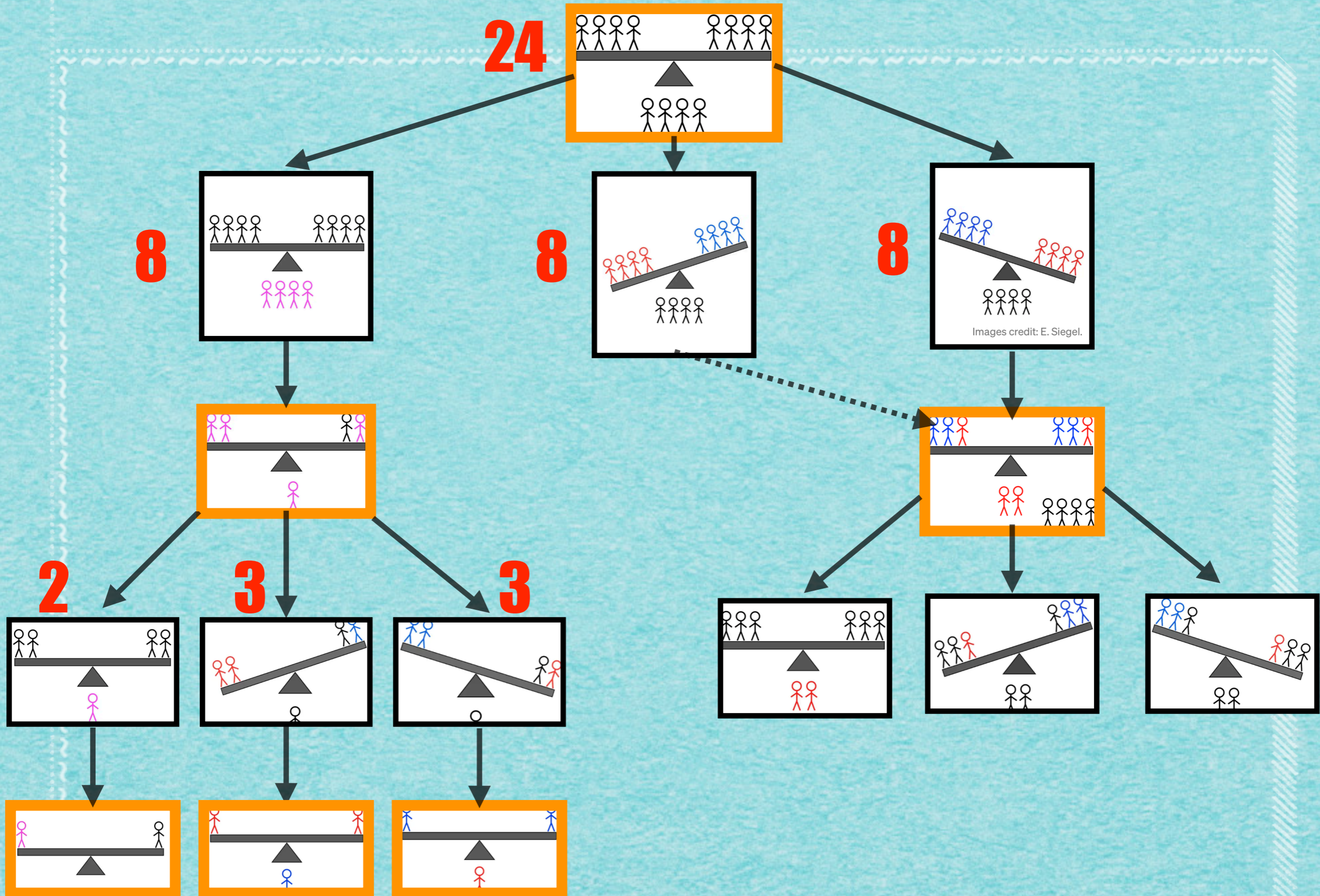
Images credit: E. Siegel.

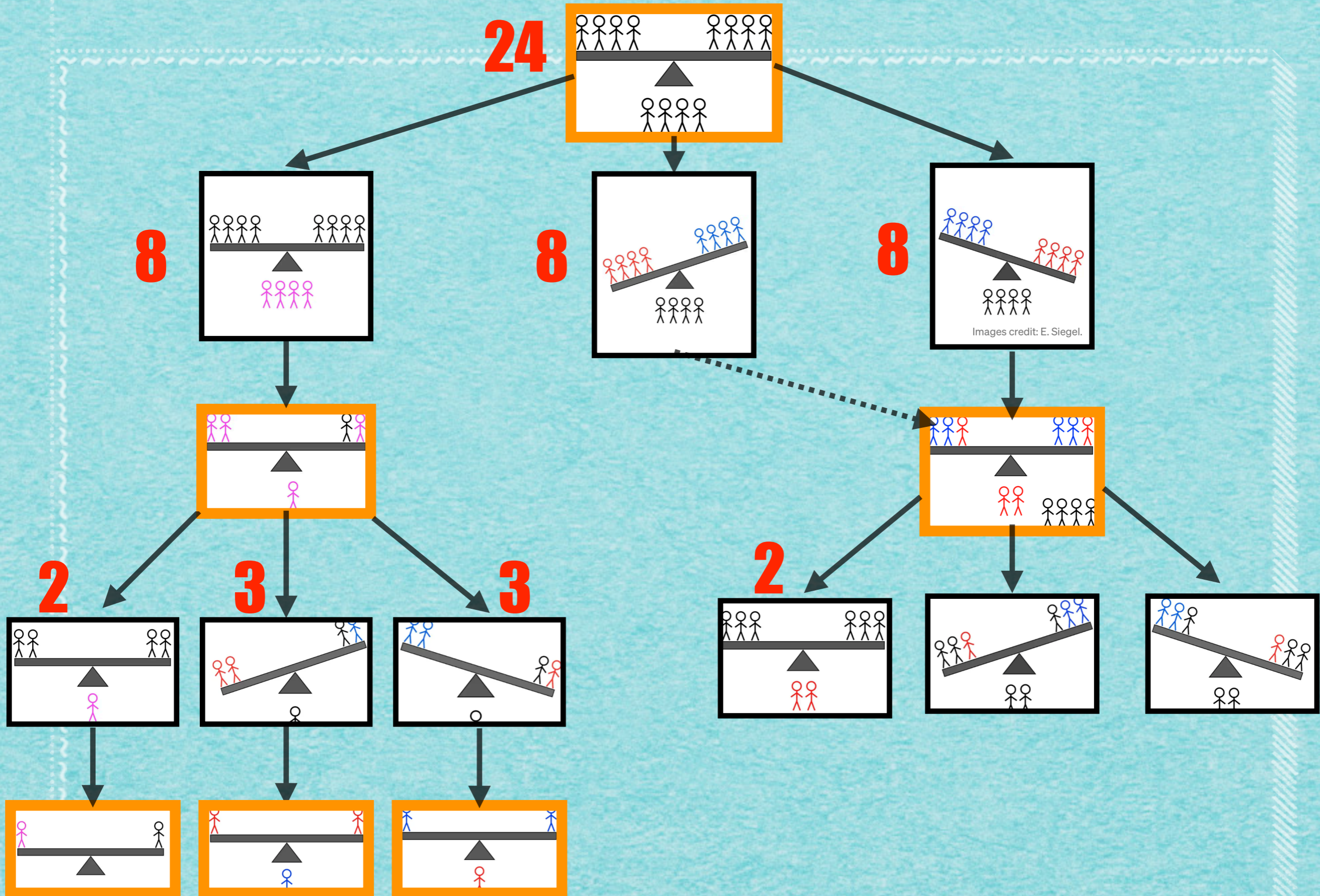


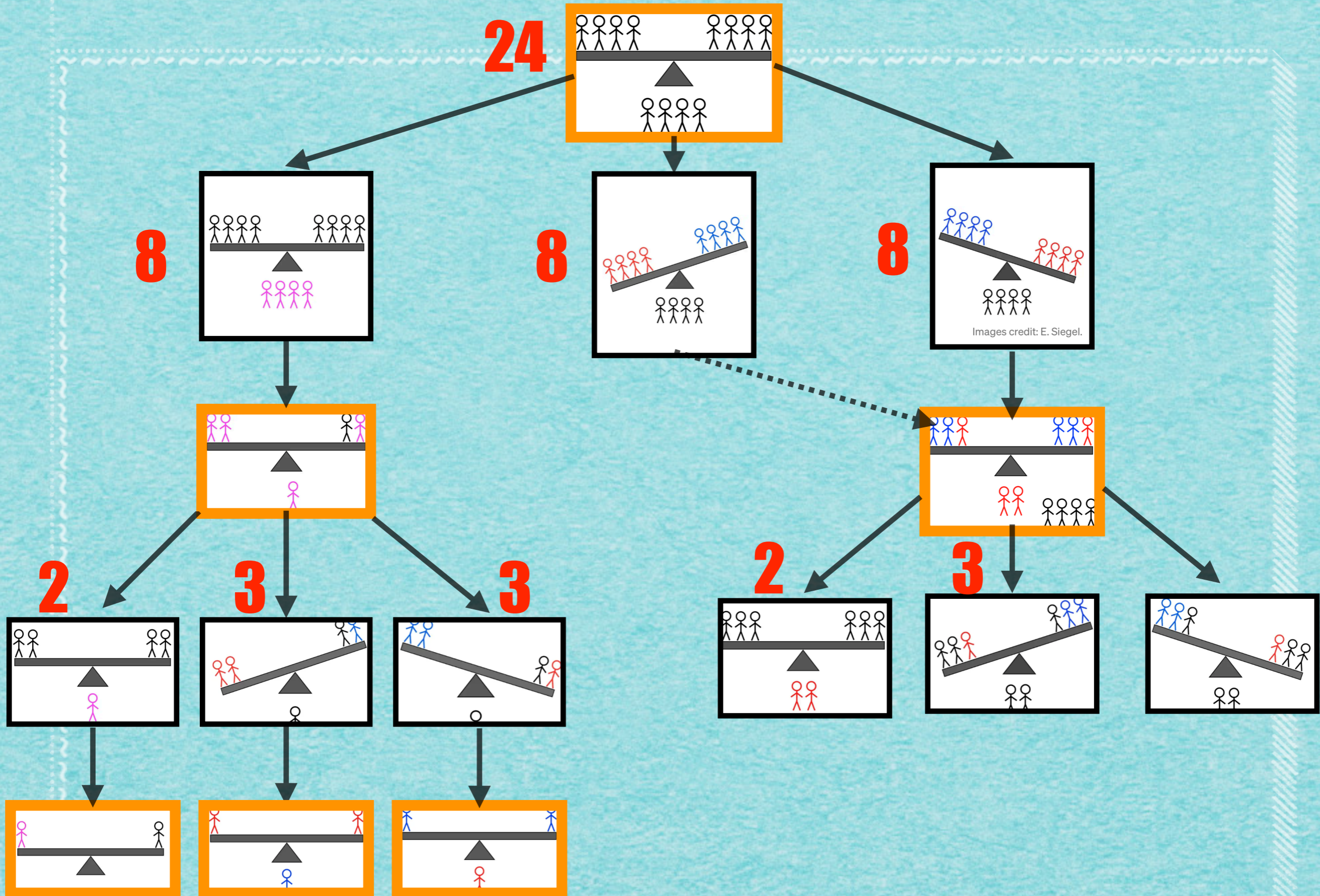


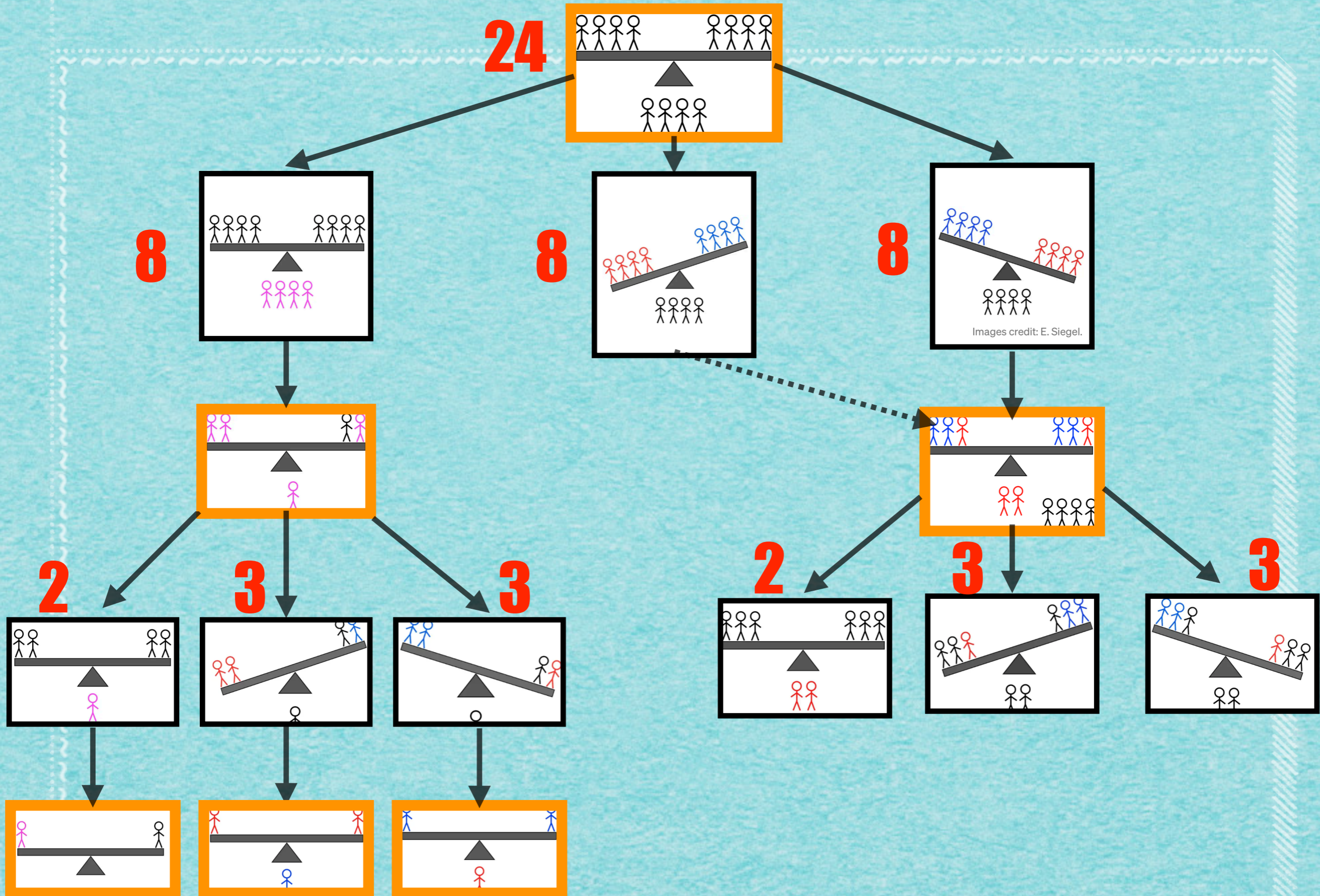
Images credit: E. Siegel.

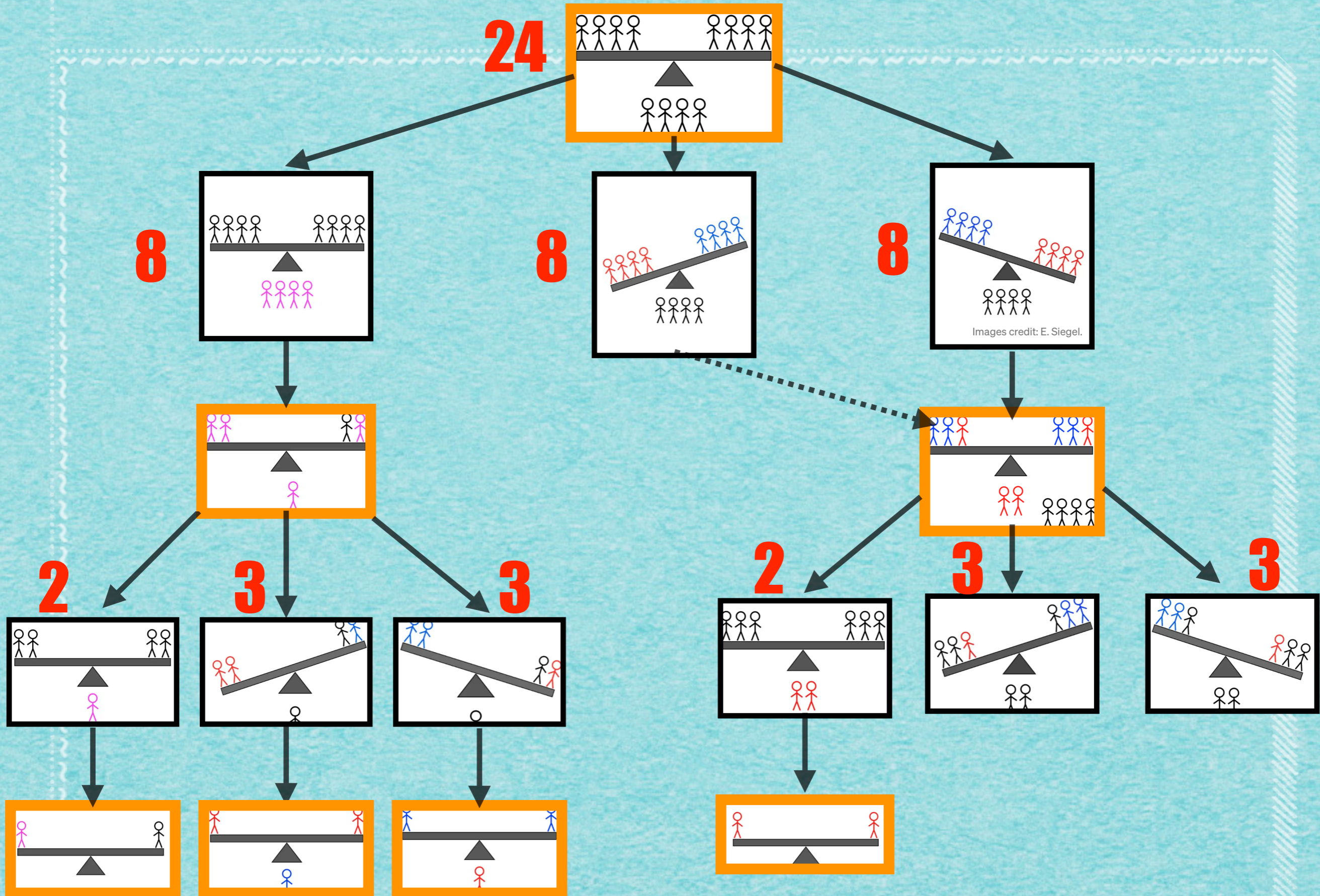


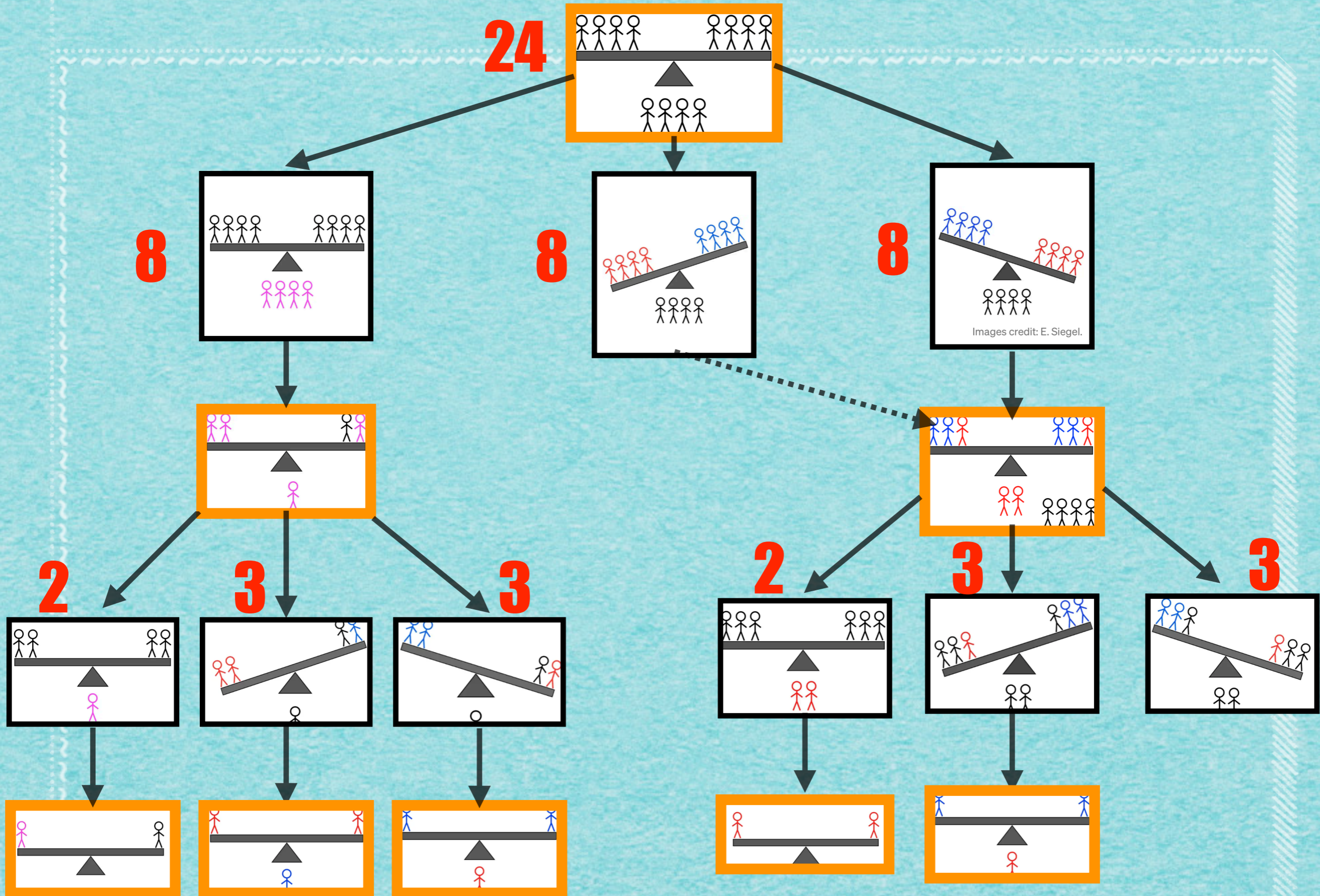


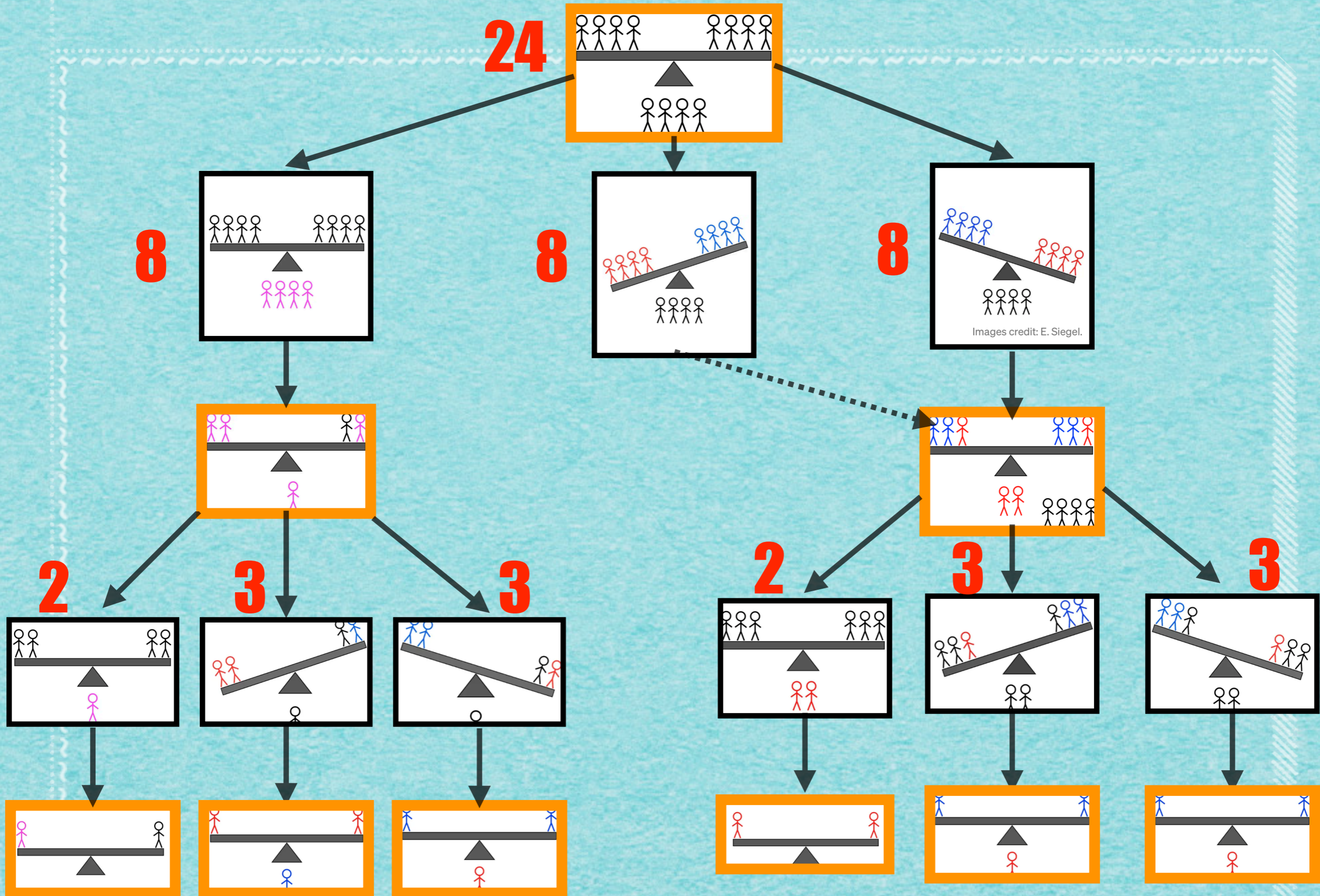












If you were able to solve it, you were able to do what Captain Holt, Amy Santiago, Terry Jeffords and the rest of the Nine-Nine couldn't do: avoid a tragic #nerdfail.



Image credit: Fox / Fremulon.



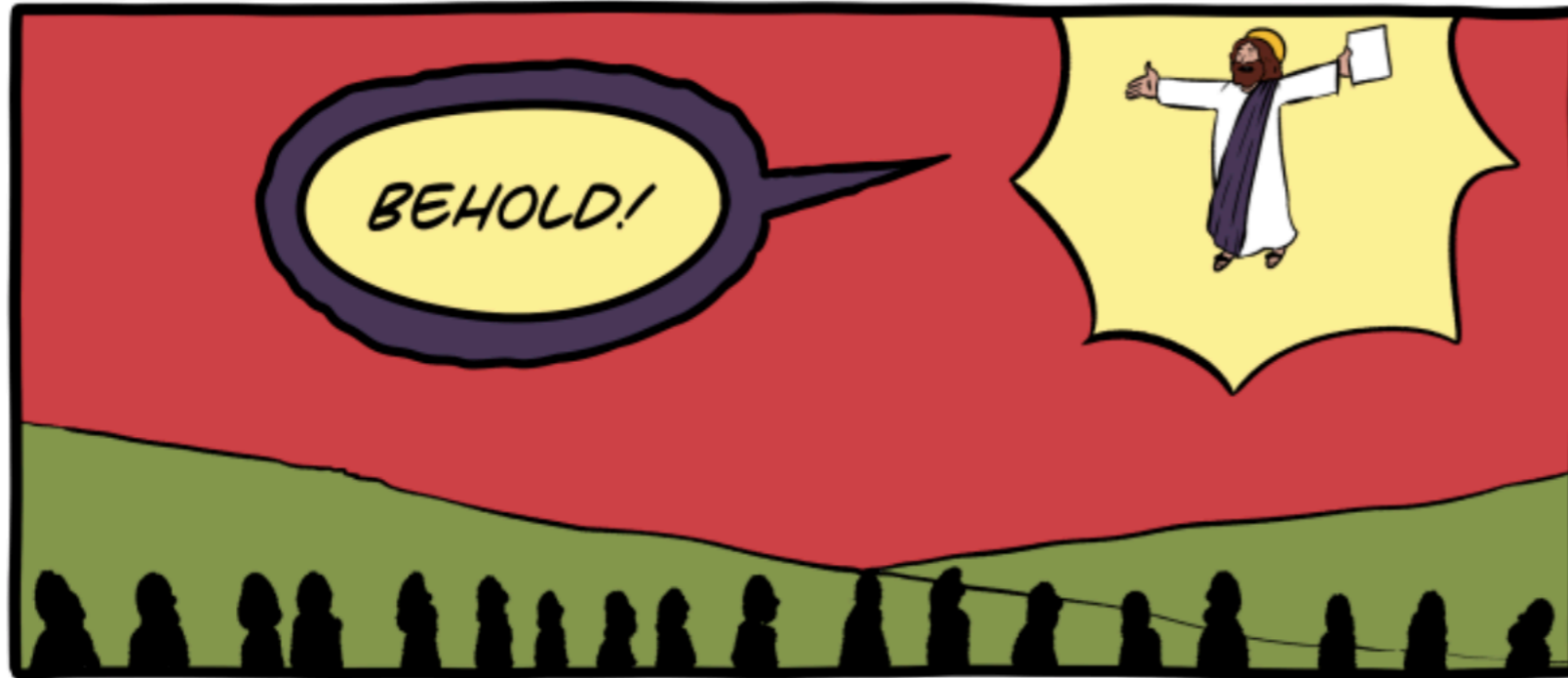
Untere Schranke von $\Omega(n \log n)$ für Sortieren



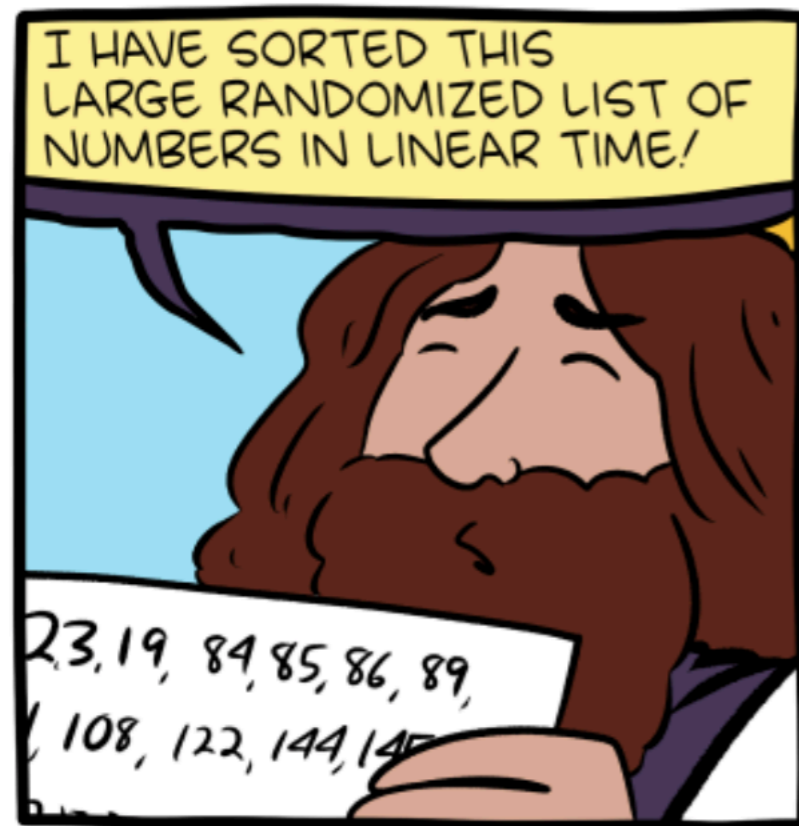
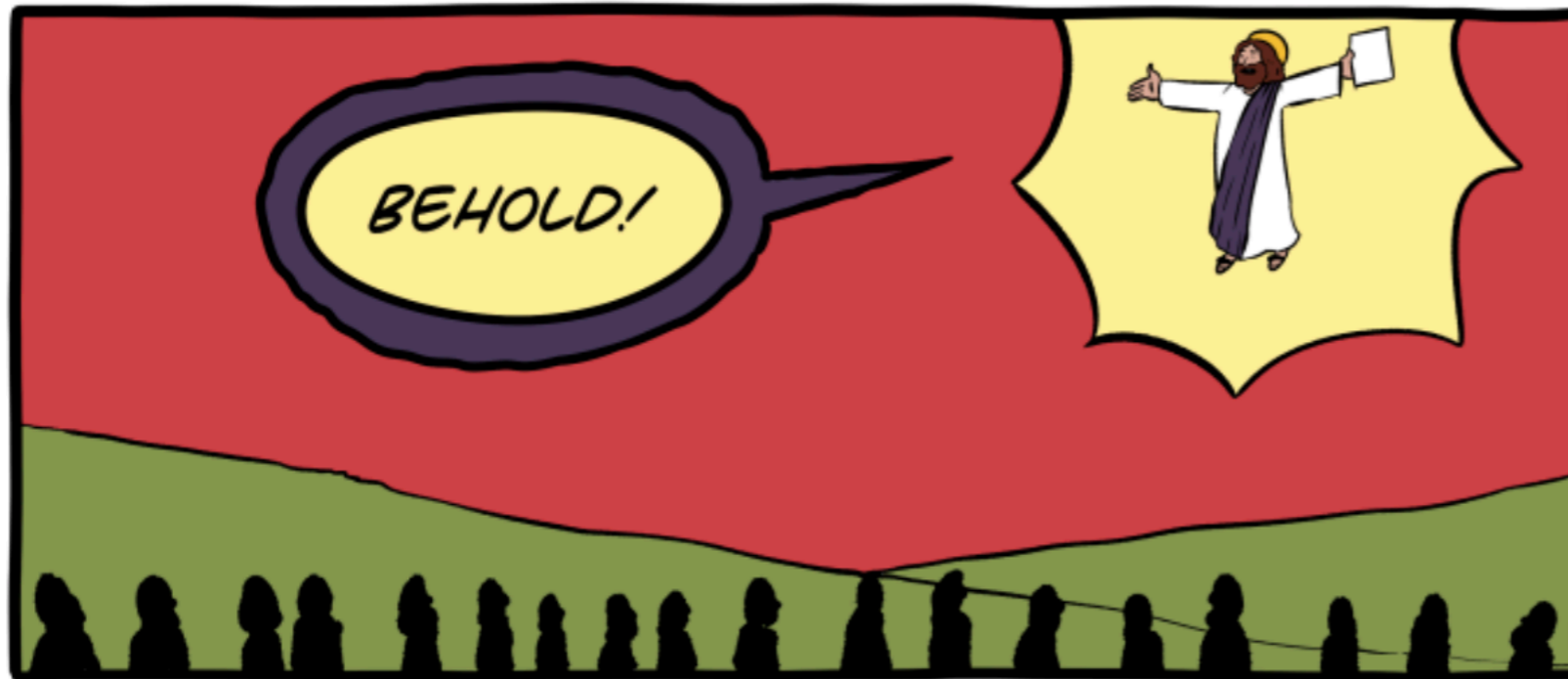
Untere Schranke von $\Omega(n \log n)$ für Sortieren



Untere Schranke von $\Omega(n \log n)$ für Sortieren



Untere Schranke von $\Omega(n \log n)$ für Sortieren



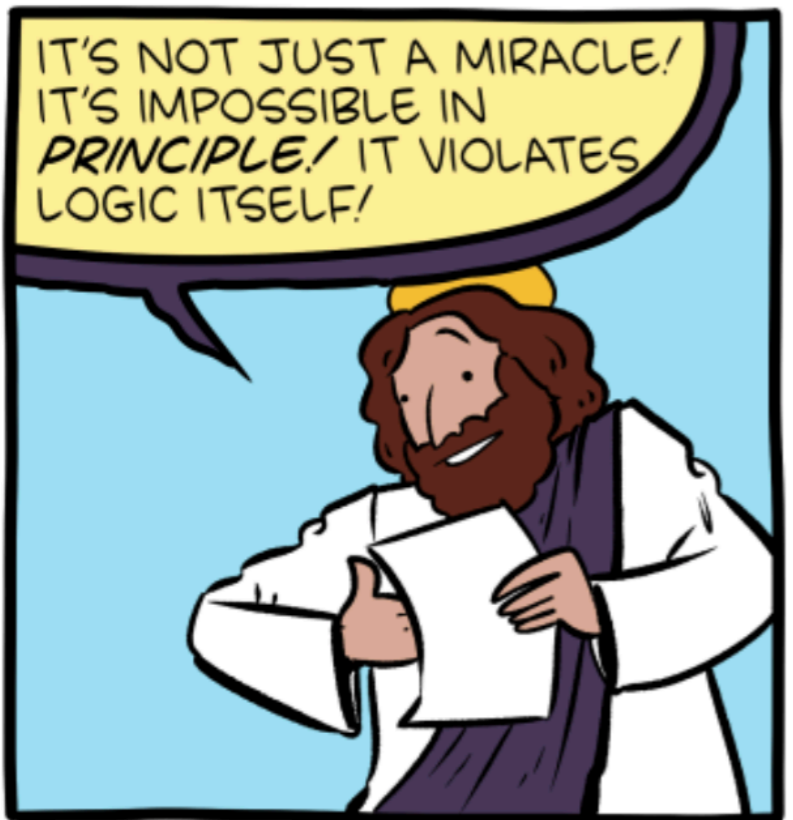
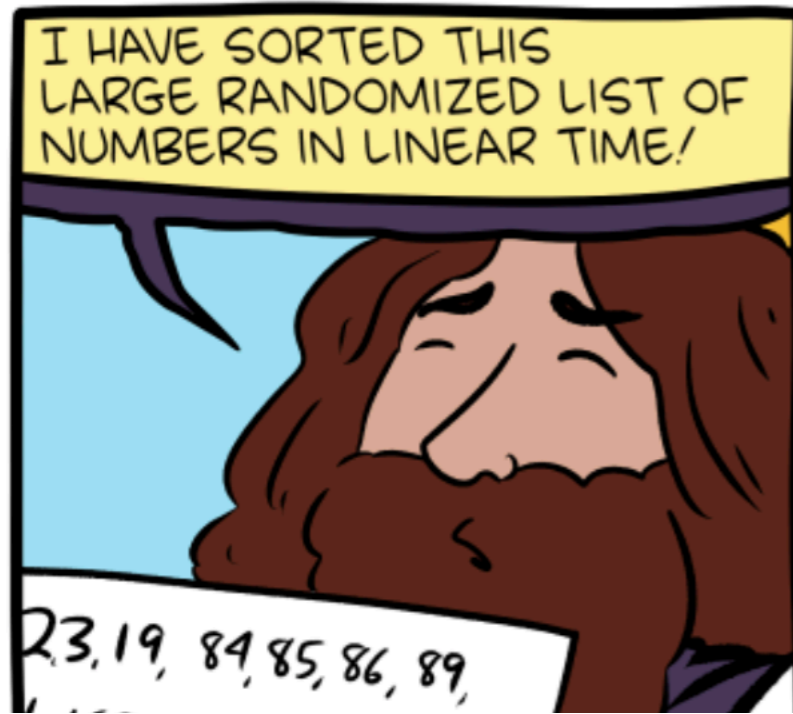
Untere Schranke von $\Omega(n \log n)$ für Sortieren



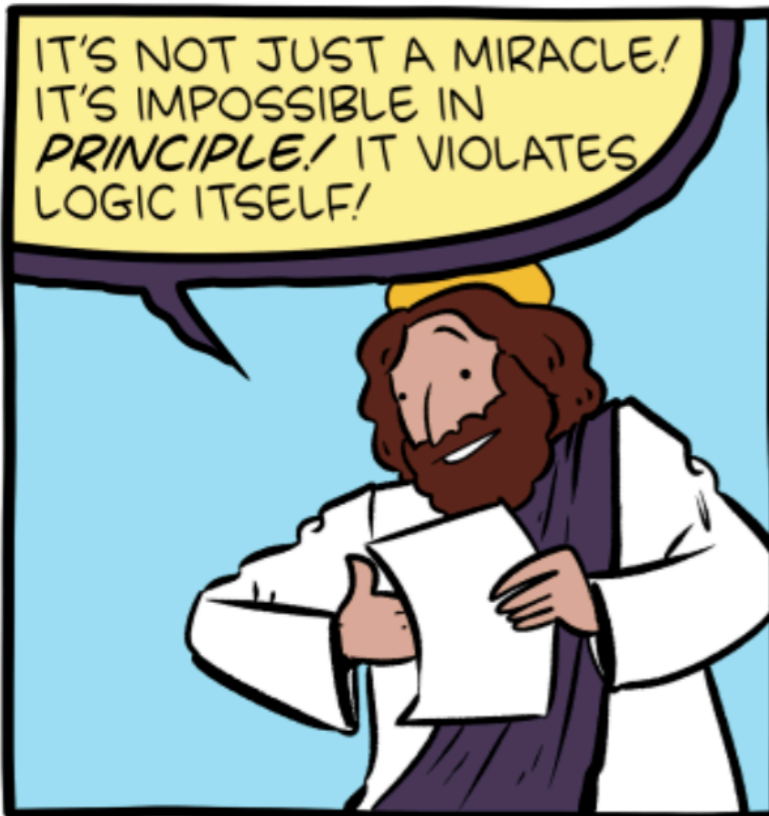
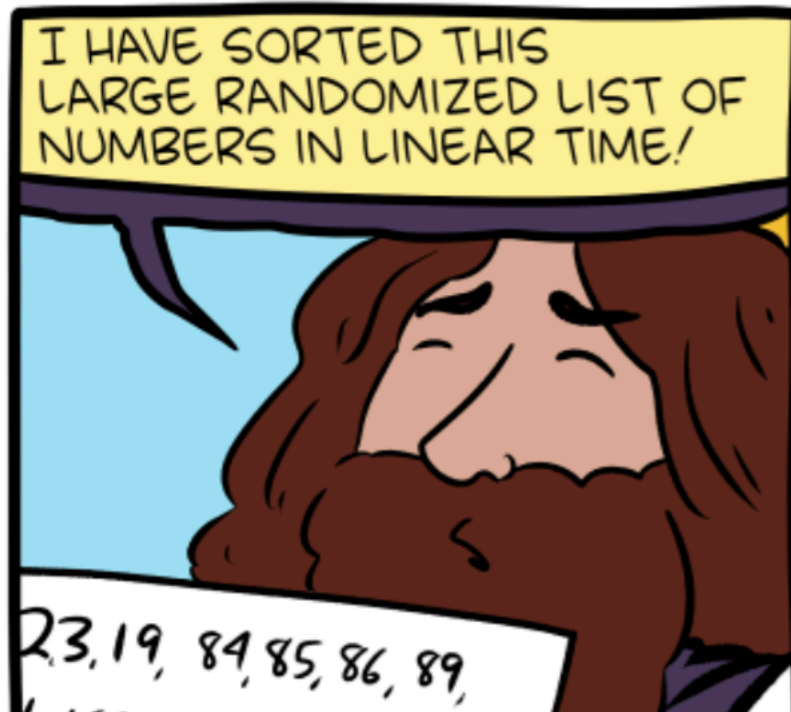
Untere Schranke von $\Omega(n \log n)$ für Sortieren



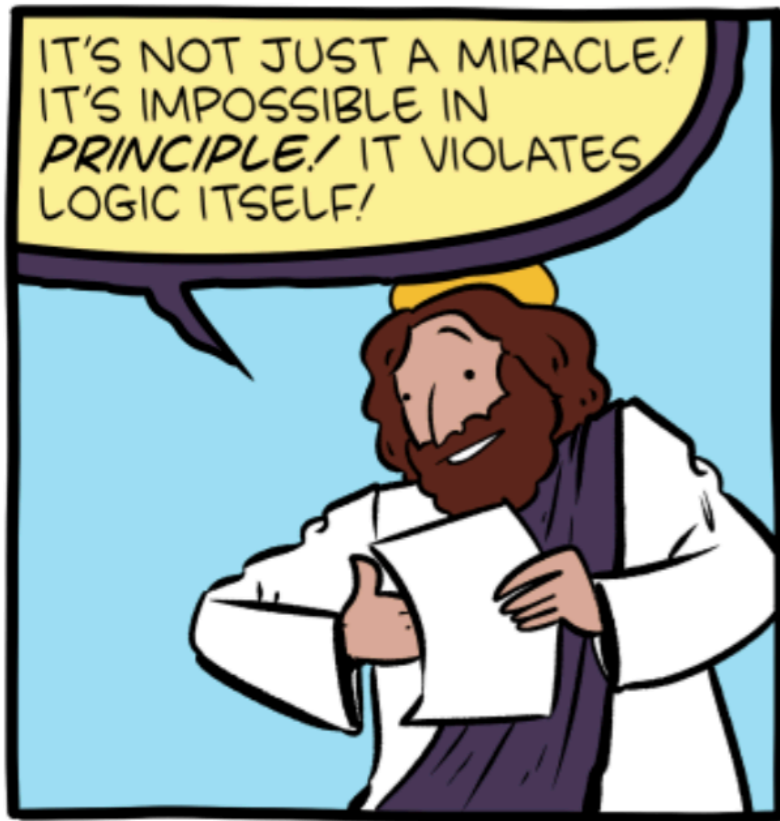
Untere Schranke von $\Omega(n \log n)$ für Sortieren



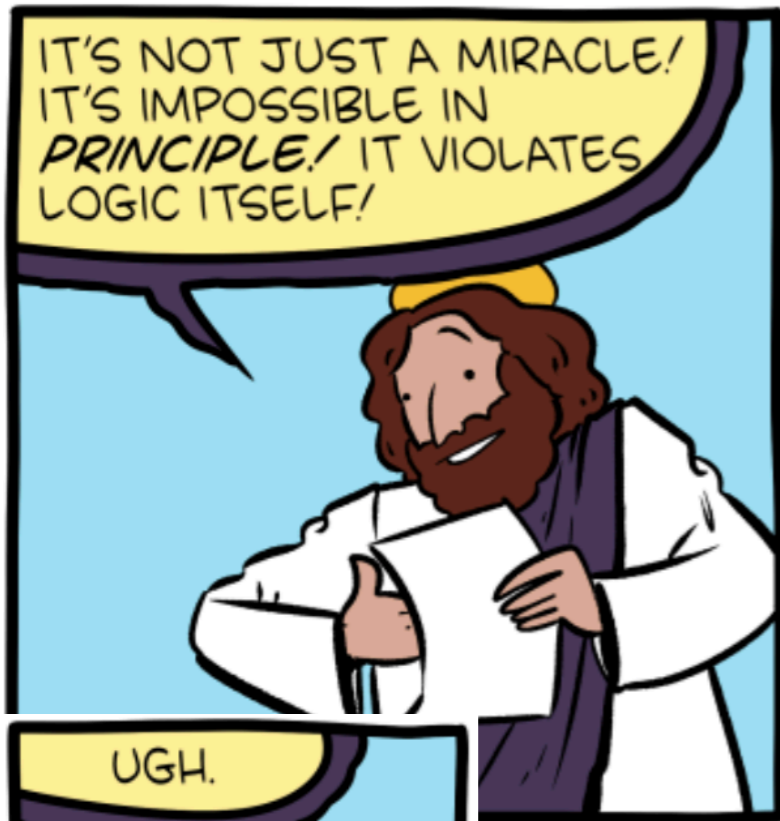
Untere Schranke von $\Omega(n \log n)$ für Sortieren



Untere Schranke von $\Omega(n \log n)$ für Sortieren



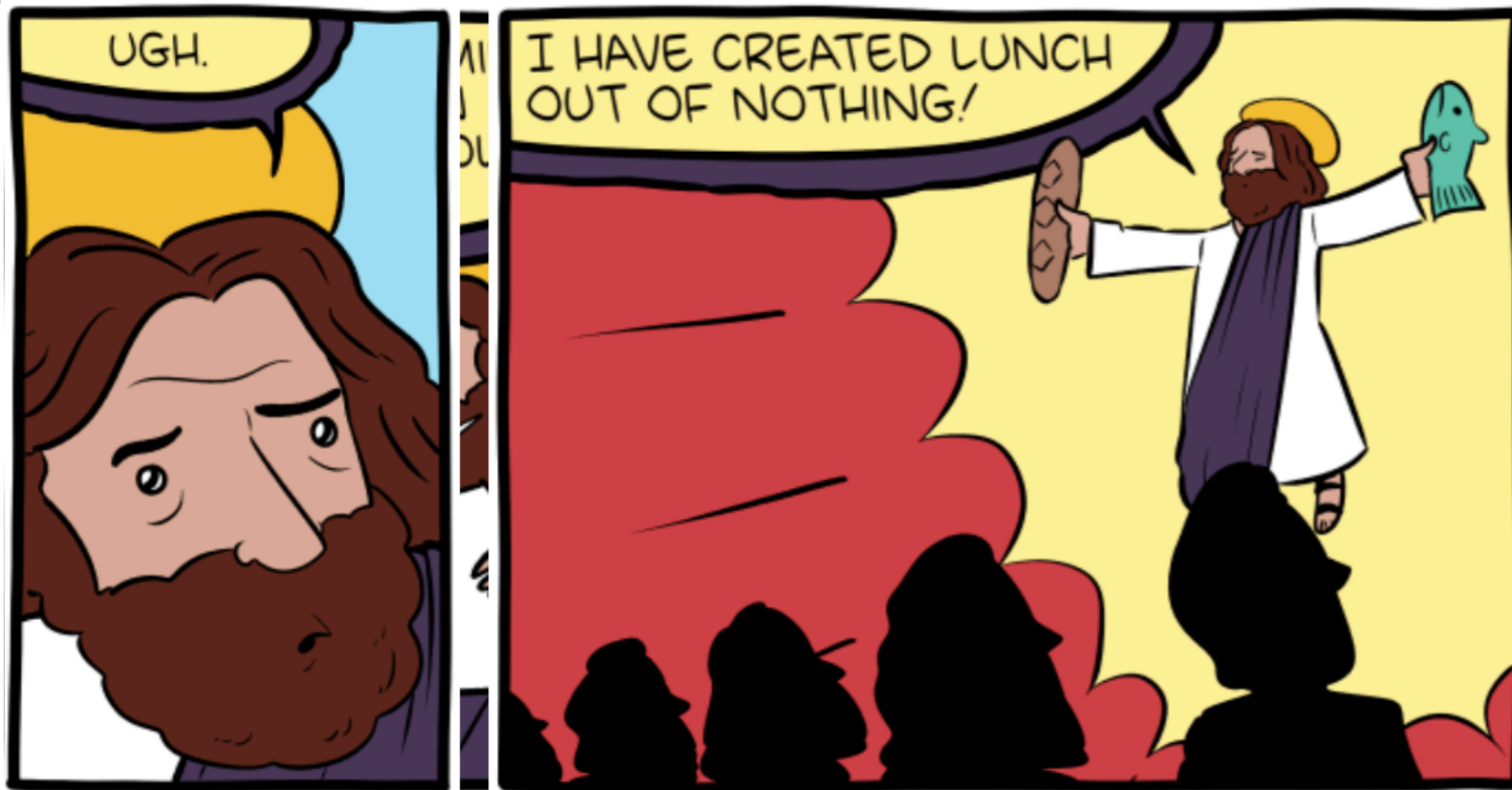
Untere Schranke von $\Omega(n \log n)$ für Sortieren



Untere Schranke von $\Omega(n \log n)$ für Sortieren



Untere Schranke von $\Omega(n \log n)$ für Sortieren



Untere Schranke von $\Omega(n \log n)$ für Sortieren



Mehr an der Tafel!

s.fekete@tu-bs.de