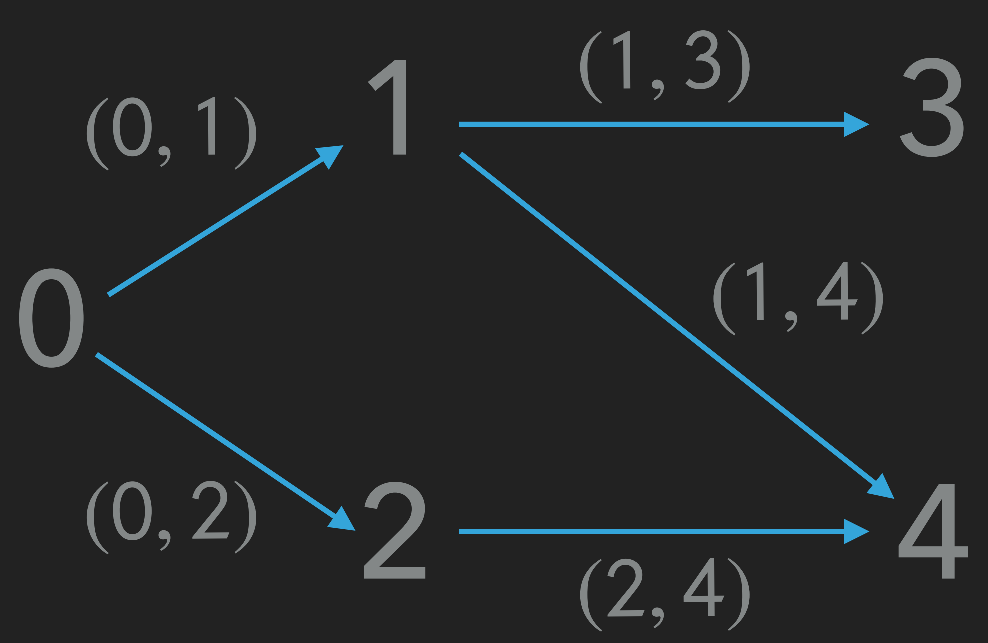
MATTEO SAMMARTINO

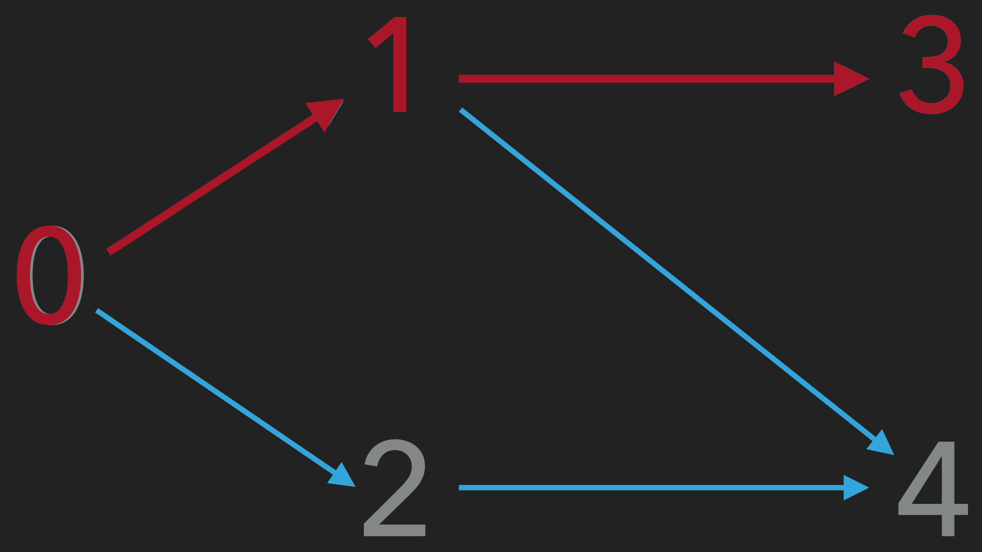
COMPUTATIONS OVER INFINITE DATA STRUCTURES

$R \subseteq X \times Y$ finite binary relation

$$(x, y) \in R$$



REACHABILITY

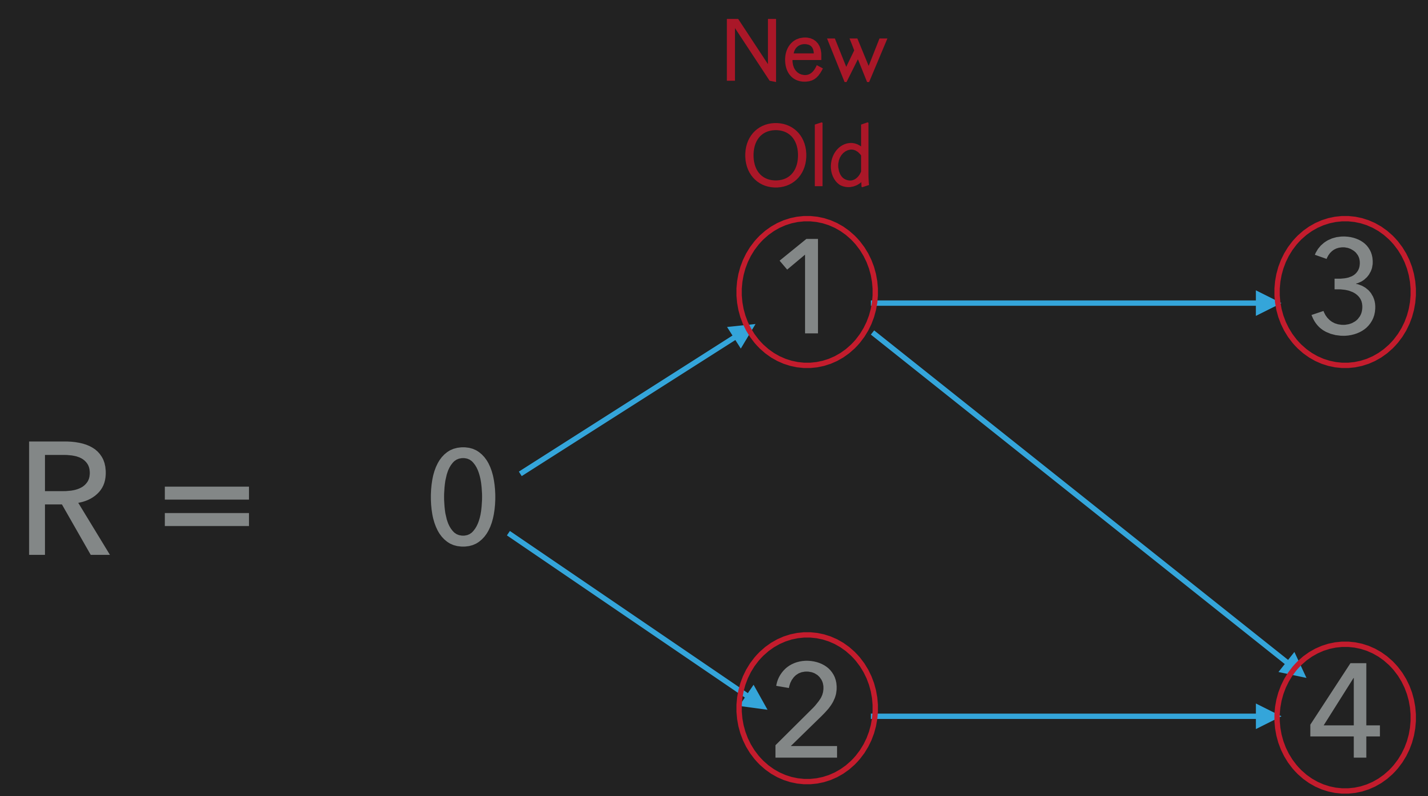


"3 is reachable from 0" = there is a path from 0 to 3

Computes elements in R reachable from S

```
function reach (R,S)
  New := S
  repeat
   Old := New
   for (x,y) in R do
     if x ∈ Old then New := Old ∪ {y}
  until Old = New
```

EXAMPLE



$S = \{1, 2\}$

```
function reach (R,S)
  New := S
  repeat
   Old := New
   for (x,y) in R do
     if x ∈ Old then New := Old ∪ {y}
  until Old = New
```

WHAT HAPPENS IF R IS INFINITE?

```
function reach (R,S)
  New := S
  repeat
   Old := New
   for (x,y) in R do
     if x ∈ Old then New := Old ∪ {y}
  until Old = New
```

iterates over infinitely many pairs

tests membership of a potentially infinite set

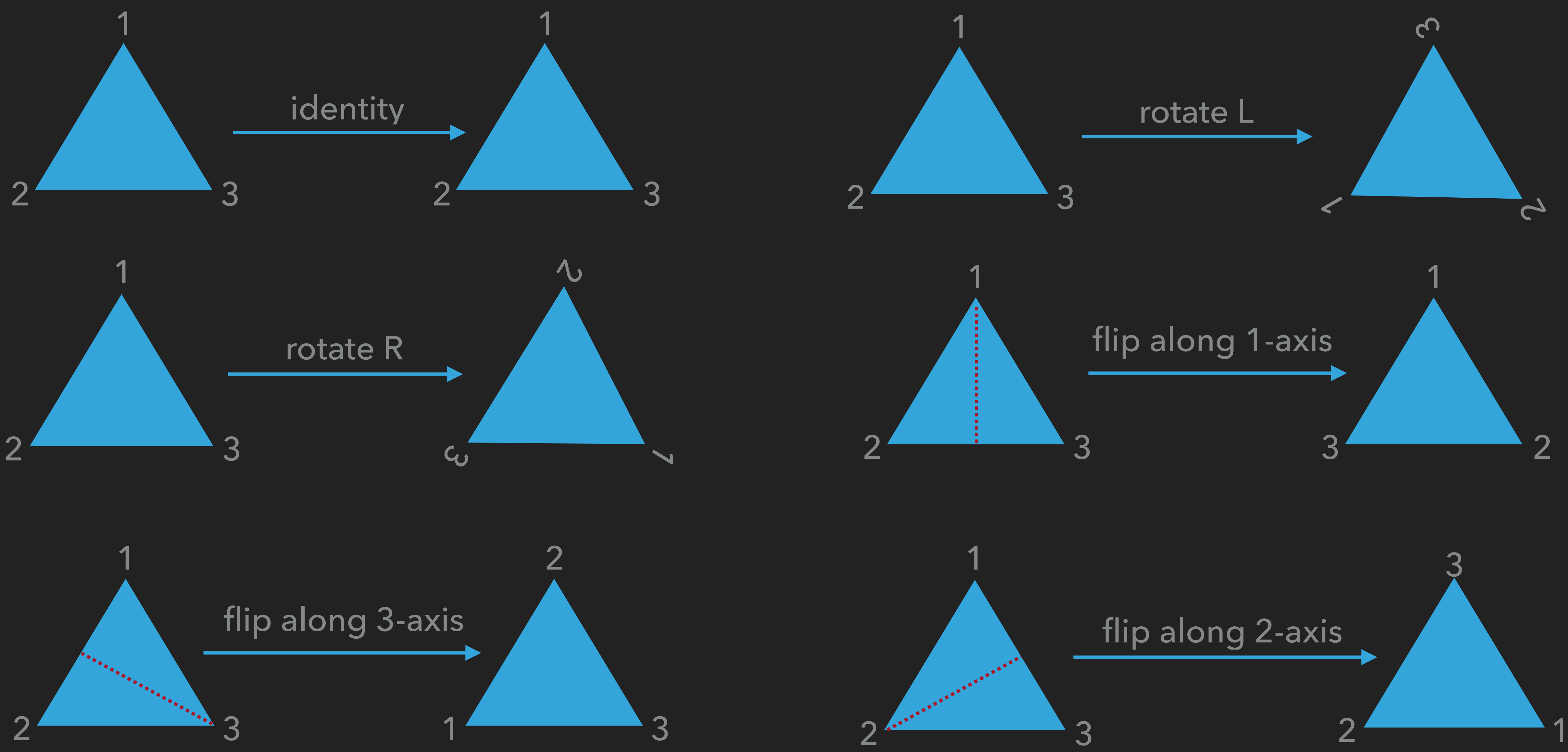
compares potentially infinite sets

THEORY CAN HELP!

THEORY OF NOMINAL SETS

SYMMETRY GROUP

“Transformations under which an object is invariant”



group under composition

DATA TRANSFORMATIONS

$\mathbb{D}$ data values

$\pi : \mathbb{D} \to \mathbb{D}$ permutations

(bijective functions)

EXAMPLE

$\mathbb{N}$

$\pi :$ $0 \to 1$, $1 \to 0$

identity elsewhere

NOMINAL SET

collection of data structures

$$(S, \cdot)$$

LEFT ACTION

$$\pi \cdot s = s'$$

permutation "acts" on data structure s

$$\mathrm{id} \cdot s = s$$

$$\pi \cdot (\pi' \cdot s) = (\pi \circ \pi') \cdot s$$

each $s \in S$ has **finite support**

$$\mathrm{supp}(s) \subseteq \mathbb{D}$$

$\forall \pi$ fixing $\mathrm{supp}(s)$

$$\pi \cdot s = s$$

DATA STORED IN s

EXAMPLE: STRINGS

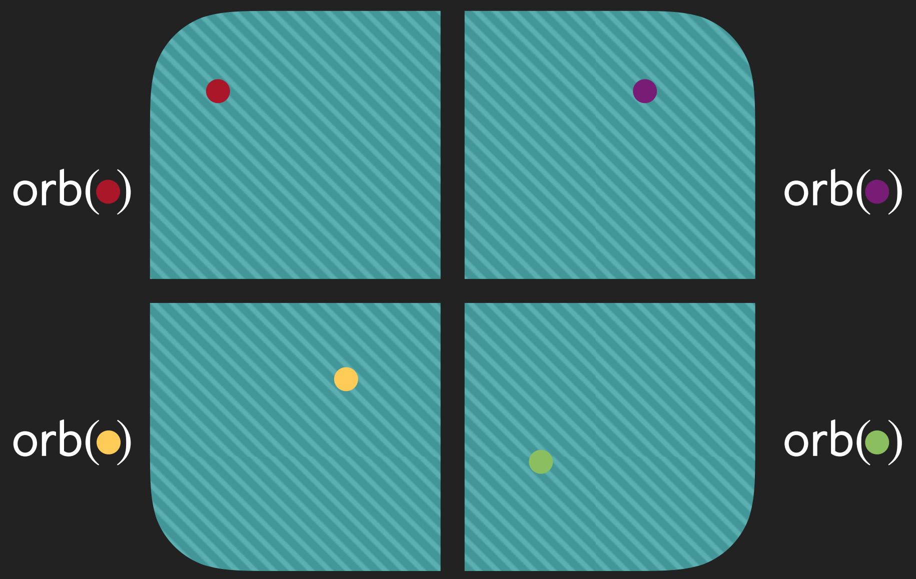
$$(Str, \cdot\ )$$

$$\pi \cdot abcd = \pi(a)\pi(b)\pi(c)\pi(d)$$

$$\mathrm{supp}(abcd) = \{a, b, c, d\}$$

ORBIT

$$\mathrm{orb}(s) = \{\pi \cdot s \mid s \in S, \pi \text{ permutation}\}$$

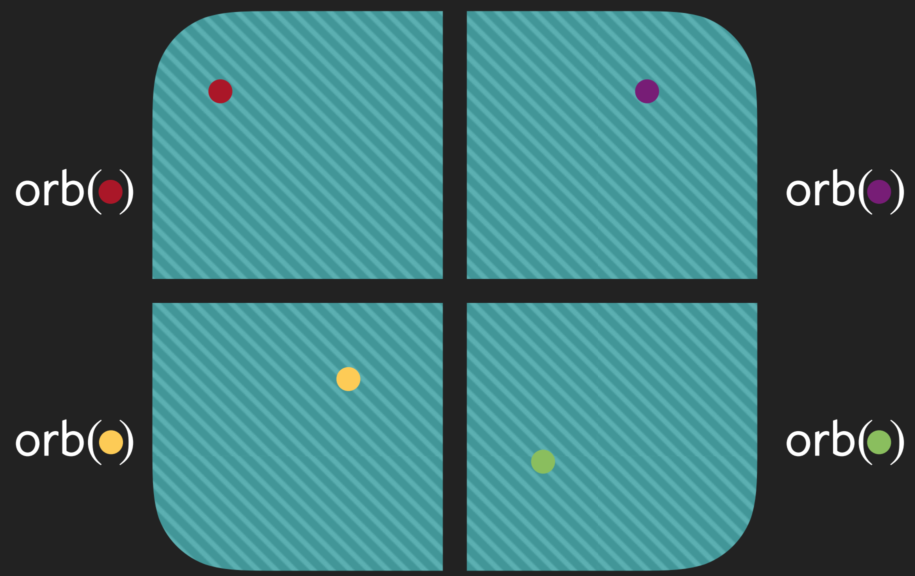
$S =$ 

orbit-finite = can be partitioned into finitely-many orbits

WE CAN COMPUTE OVER ORBIT-FINITE SETS!

$$\mathrm{algorithm}(I_1, \ldots, I_n) = O$$

$I_1, \ldots, I_n, O$ **orbit-finite nominal sets**



$x \in \{\bullet, \bullet, \bullet, \bullet\}$

for $\cancel{x \in S}$ do

REACHABILITY

R ~~finite~~ orbit-finite binary relation

reachable elements form an orbit-finite set

R orbit-finite ~~finite~~ binary relation

S orbit-finite set

```
function reach (R,S)
  New := S
  repeat
   Old := New
   for (x,y) in R do
     if x ∈ Old then New := Old ∪ {y}
  until Old = New
```

🙂

A LOT OF COOL STUFF!

PROGRAMMING LANGUAGES

LOIS (C++) https://www.mimuw.edu.pl/~erykk/lois/

NLAMBDA (HASKELL) https://www.mimuw.edu.pl/~szynwelski/nlambda/

AUTOMATA THEORY

Mikolaj Bojanczyk, Bartek Klin, Slawomir Lasota:
Automata Theory in Nominal Sets. Logical Methods in Computer Science 10(3) (2014)

AUTOMATA LEARNING ALGORITHM

Joshua Moerman, Matteo Sammartino, Alexandra Silva, Bartek Klin, Michal Szynwelski:
Learning nominal automata. POPL 2017: 613-625

"SIGHTLY INFINITE SETS"

https://www.mimuw.edu.pl/~bojan/upload/main-7.pdf
(continuously updated)