Escuela de Investigación CIMPA

"Álgebra, Combinatoria y Física"

Facultad de Ingeniería, Universidad de Valparaíso, Chile

Valparaíso, 20-31 Enero 2014 -

29 Enero 2014

curso I

Xavier Viennot

LaBRI, CNRS, Bordeaux

http://cours.xavierviennot.org

Courses Talca, Chile 2013/2014

Cours Universidad de Talca

(December 2013 - January 2014)

Heaps of pieces (24 h)

(with interactions in mathematics and physics)

Basic definitions and theorems:

Ch1 Commutations monoids and heaps of pieces: basic definitions

Ch2 Generating functions for heaps of pieces

Ch3 Heaps and paths, flow monoids, rearrangements

Some applications in classical mathematics:

Ch4 Heaps and linear algebra: bijective proofs of classical theorems

Ch5 Heaps and combinatorial theory of orthogonal polynomials and continued fractions

Ch6 Heaps and algebraic graph theory

Some applications in theoretical physics:

Ch7 Directed animals and gas model in statistical physics, Lorentzian triangulations in 2D quantum gravity

Ch8 Polyominoes, q-analogue and SOS model in physics

Applications to more advanced mathematics:

Ch9 Fully commutative class of words in Coxeter groups

[Representation theory in Lie algebras with operators on heaps] http://cours.xavierviennot.org

Courses Talca, Chile 2013/2014

Cours Universidad de Talca

(December 2013 - January 2014)

Heaps of pieces (24 h)

(with interactions in mathematics and physics)

X.G.Viennot,
Introduction to the theory of pieces with applications to statistical mechanics and quantum gravity

in workshop *"Combinatorial Identities & their Applications in Statistical Mechanics"*,

Isaac Newton Institute for Mathematical Science, Cambridge, 7 April 2008 slides and video

Ch 1

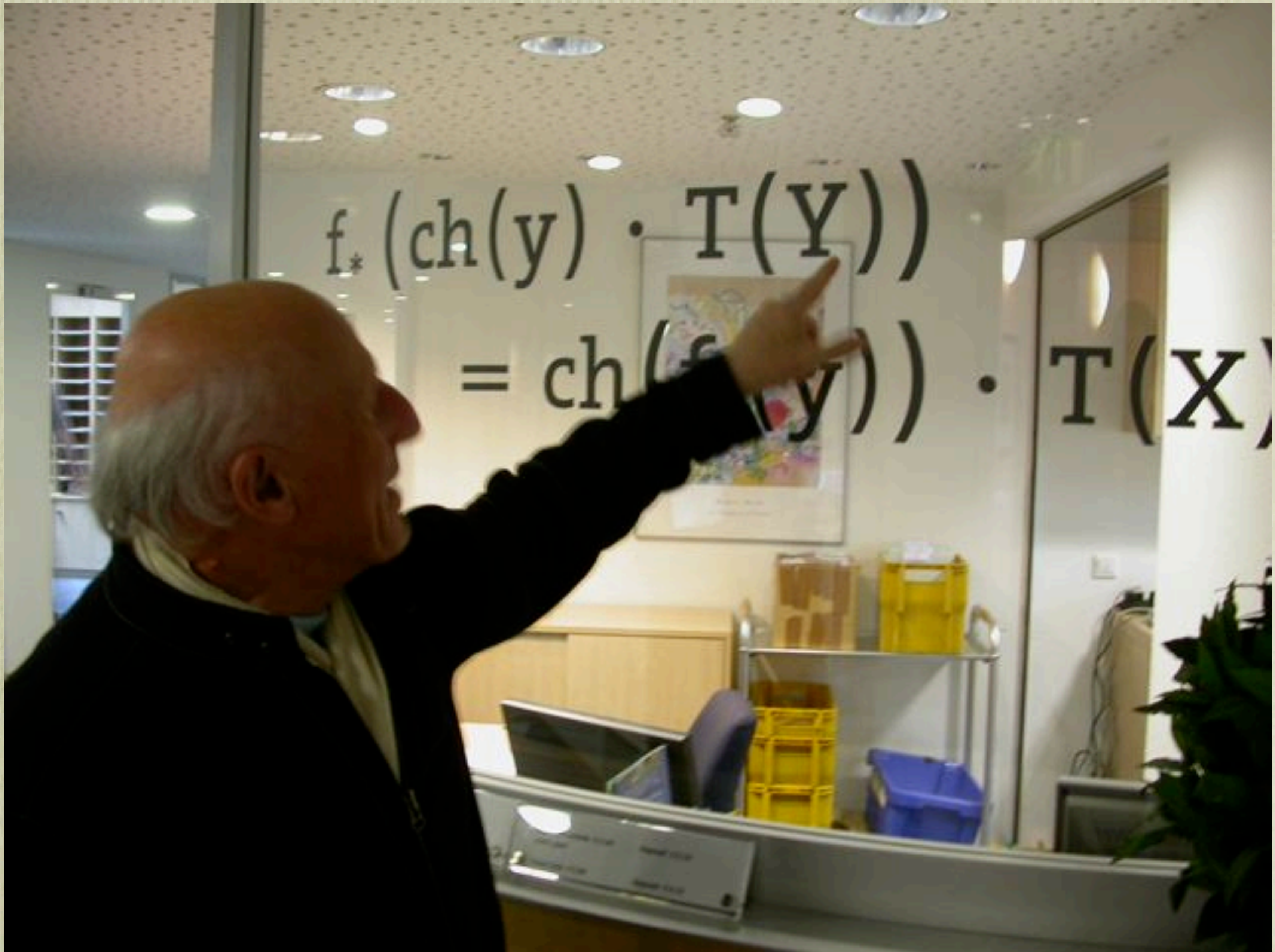
Commutation monoids and heaps of pieces: basic definitions

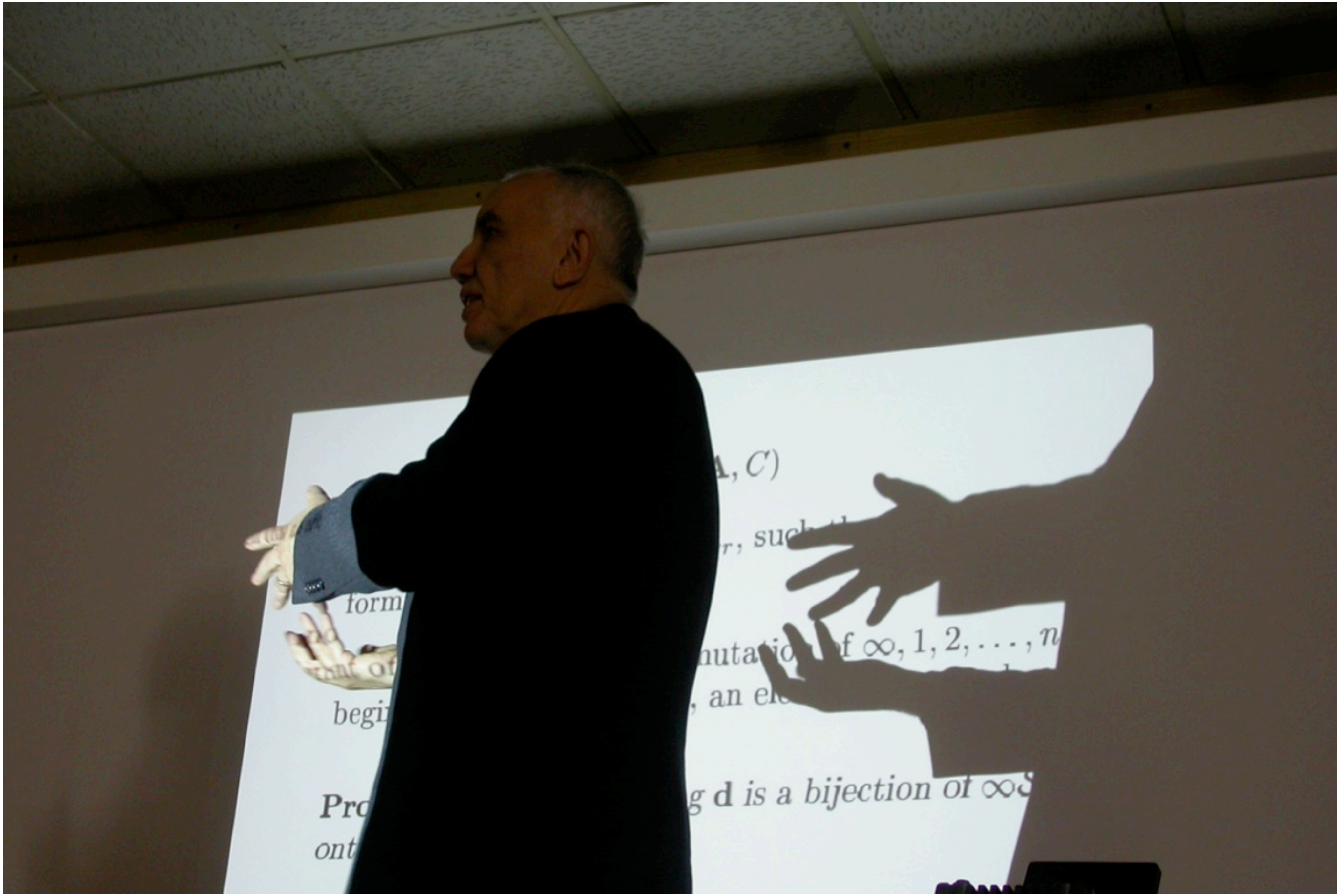
§1 Commutation monoids

Cartier-Foata **commutation** **monoid**

Lecture Note in Maths n°85 (1969)

"Problèmes combinatoires de commutation et réarrangements"


$f_*(ch(y) \cdot T(Y))$


$\infty, 1, 2, \ldots, n$
d is a bijection of

alphabet A

free monoid A^*

words $w = a_1 a_2 \cdots a_p$

product : concatenation

$$\left.\begin{array}{l} u = a_1 \cdots a_p \\ v = b_1 \cdots b_q \end{array}\right\} uv = a_1 \cdots a_p b_1 \cdots b_q$$

empty word

commutation relation C antireflexive symmetric

$\equiv_C$ congruence of A^* generated by the commutations

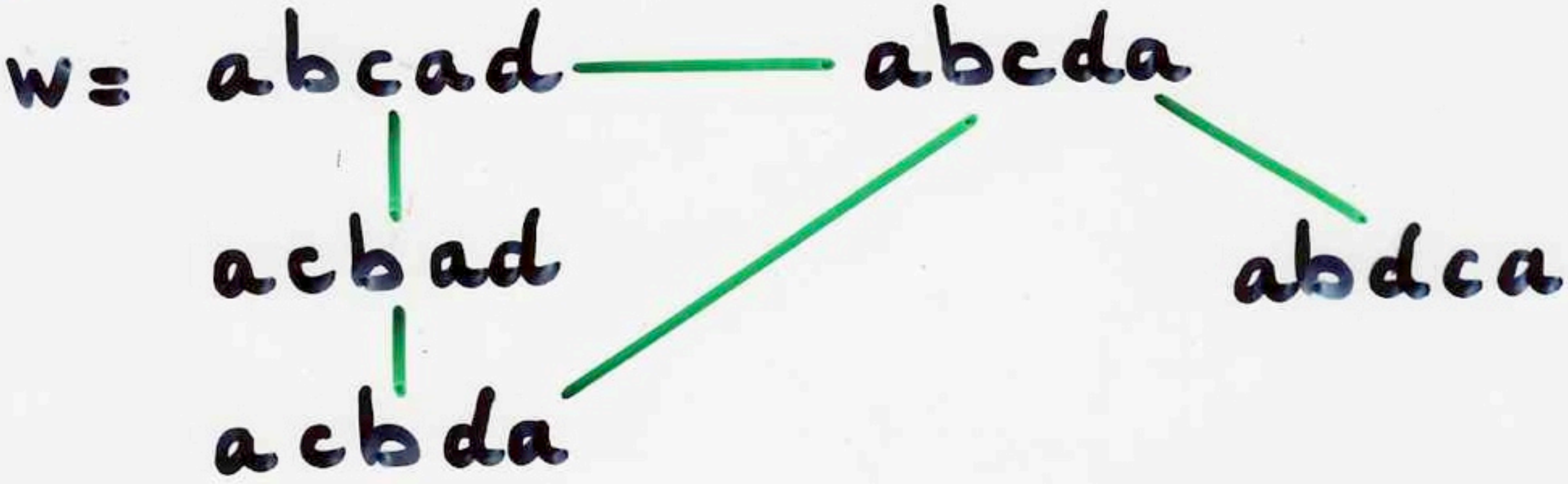
$ab \equiv ba$ iff $a C b$

<u>ex</u> :

$A = \{a, b, c, d\}$

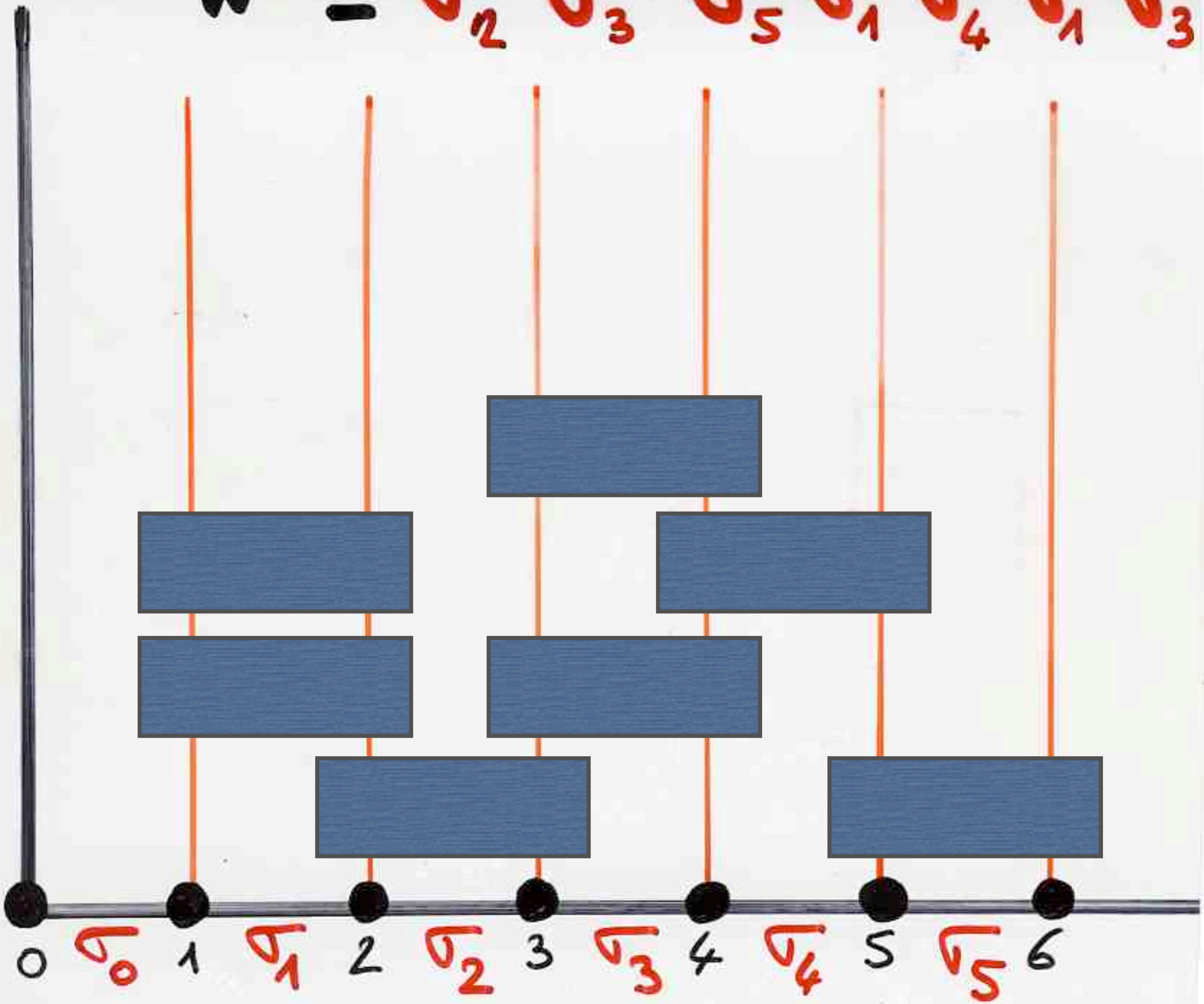
$C \quad \begin{cases} ad = da \\ bc = cb \\ cd = dc \end{cases}$

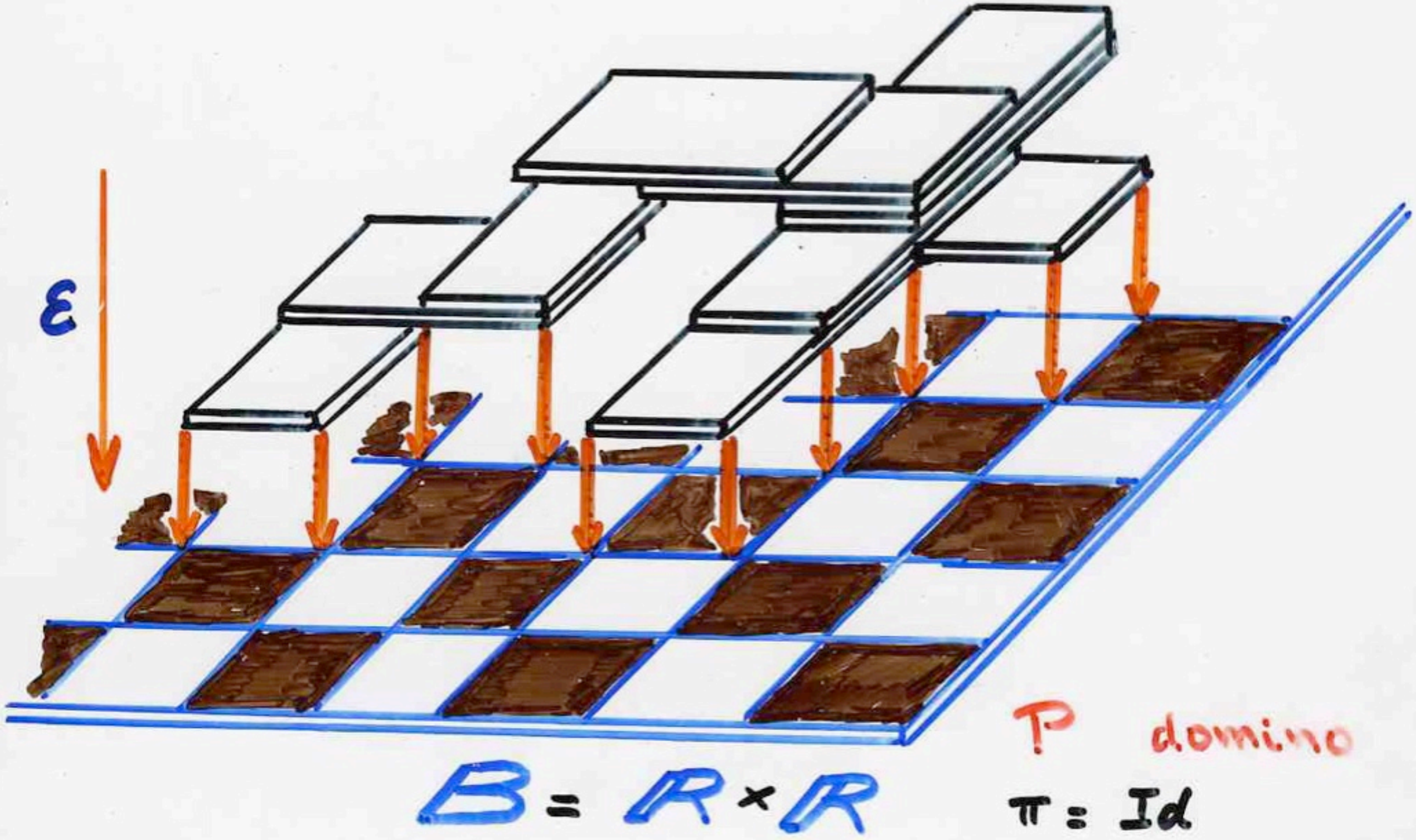
equivalence class



§2 Heaps of pieces:
basic definitions

$$w = \sigma_2\,\sigma_3\,\sigma_5\,\sigma_1\,\sigma_4\,\sigma_1\,\sigma_3$$




ε
P domino
B = ℝ × ℝ
π = Id

heap definition

- $\mathbf{P}$ set (of basic pieces)
- $\mathcal{C}$ binary relation on $\mathbf{P}$ $\begin{cases} \text{symmetric} \\ \text{reflexive} \end{cases}$

  (dependency relation)
- heap E, finite set of pairs

  (α, i) $\alpha \in \mathbf{P}$, $i \in \mathbb{N}$ (called pieces)

  projection level

(i)

(ii)

heap definition

- $\mathbf{P}$ set (of basic pieces)
- $\mathcal{C}$ binary relation on $\mathbf{P}$ $\begin{cases} \text{symmetric} \\ \text{reflexive} \end{cases}$

  (dependency relation)
- heap E, finite set of pairs

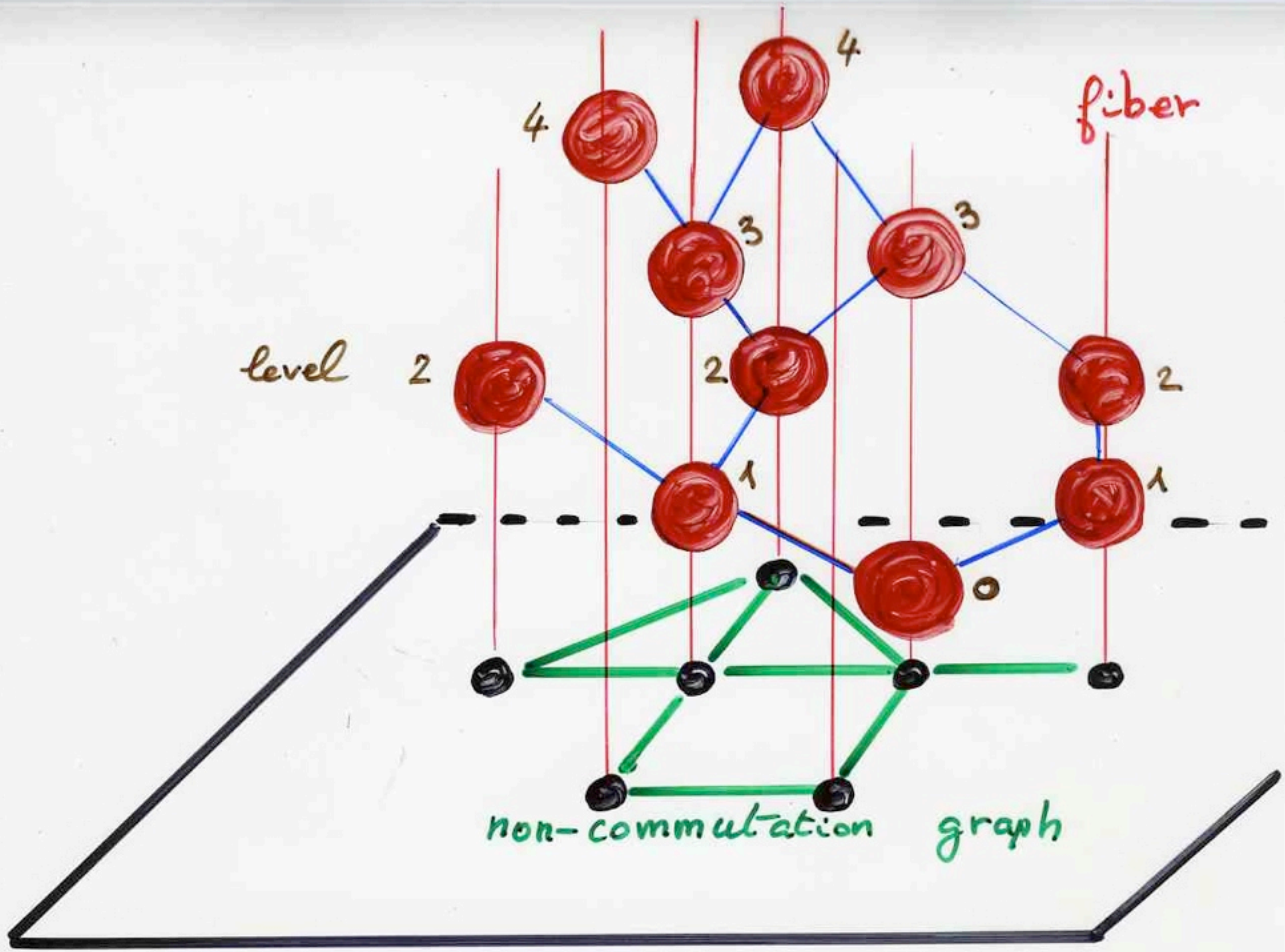
  (α, i) $\alpha \in \mathbf{P}$, $i \in \mathbb{N}$ (called pieces)

  projection level

(i) $(\alpha, i), (\beta, j) \in E$, $\alpha \,\mathcal{C}\, \beta \implies i \neq j$

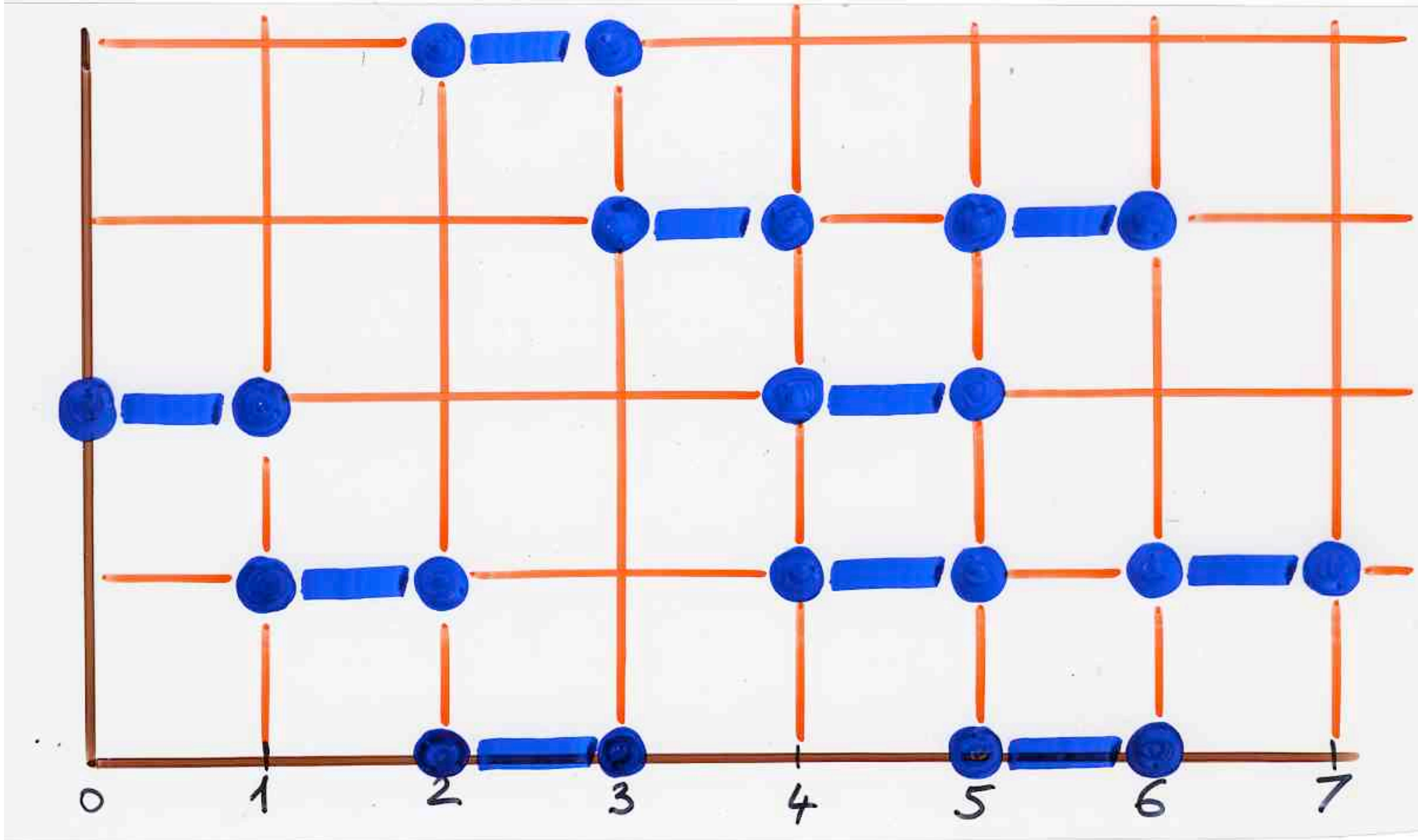
(ii) $(\alpha, i) \in E$, $i > 0 \implies \exists \beta \in \mathbf{P}$, $\alpha \,\mathcal{C}\, \beta$,

$(\beta, i-1) \in E$

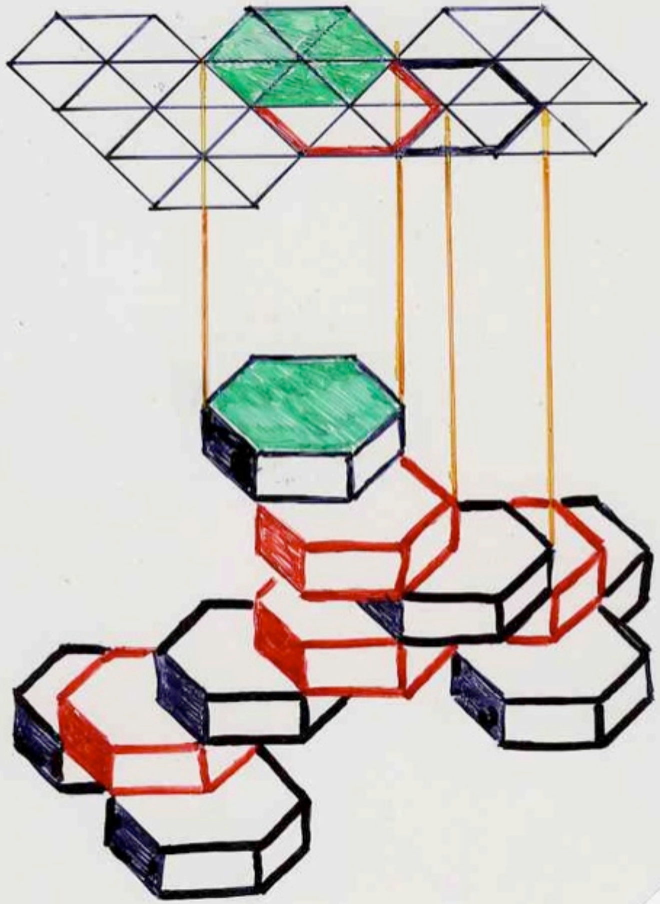


fiber
4
4
3
3
level
2
2
2
1
1
0
non-commutation graph

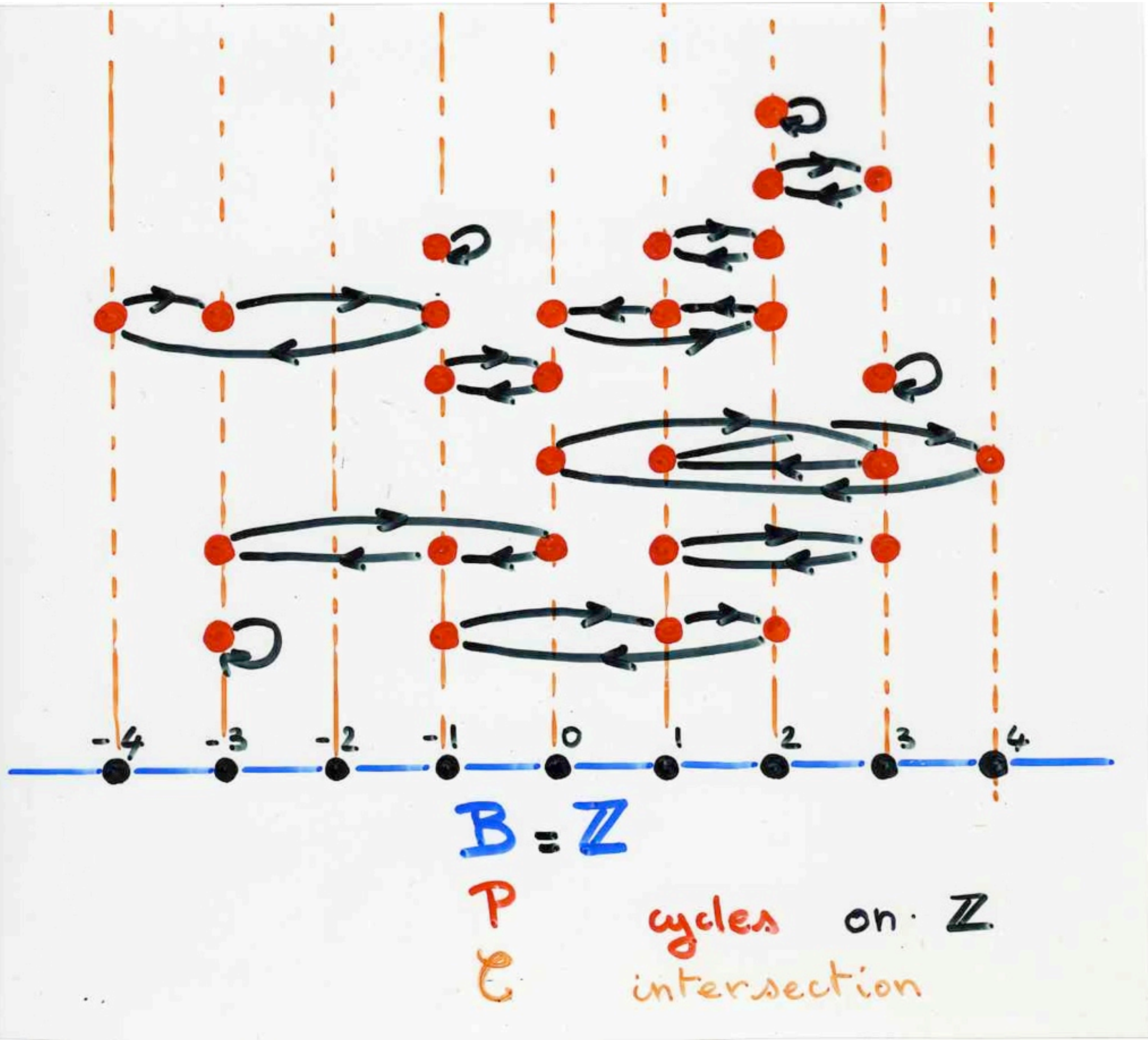
Heap of dimers over [1, n]



$$-p(-t) = y$$



KC.



-4
-3
-2
-1
0
1
2
3
4
B = Z
P cycles on · Z
C intersection

Proposition

$$\mathrm{Heap}(P, \mathcal{C}) \simeq P^{*}/\equiv_{C}$$

commutation monoid

$$C = \overline{\mathcal{C}}$$

complementary relation

heaps of **dimers**

$(i, i+1)$

on $\{0, 1, \ldots, n-1\}$

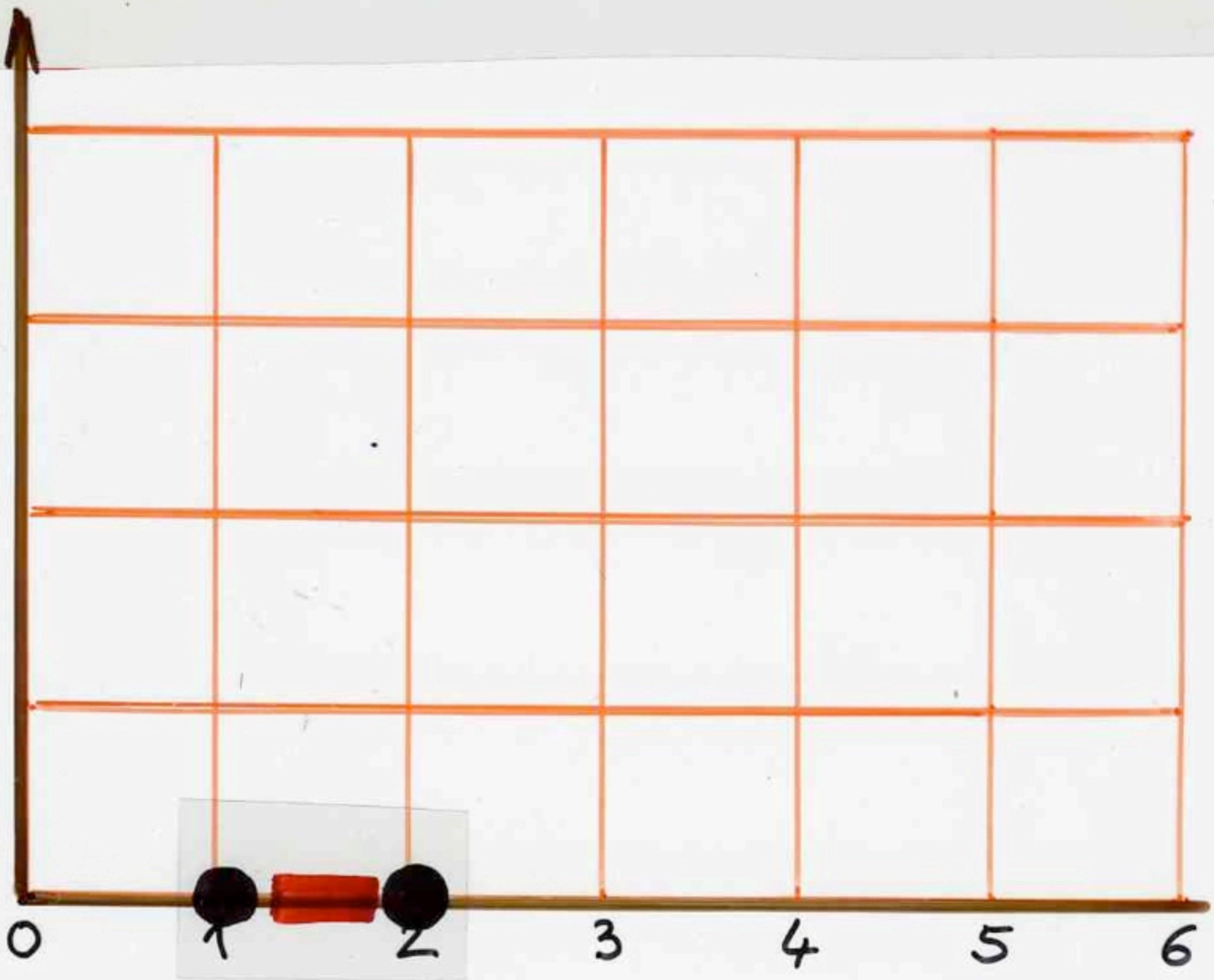
generators $\{\sigma_0, \sigma_1, \ldots, \sigma_{n-1}\}$

$$\sigma_i \sigma_j = \sigma_j \sigma_i$$

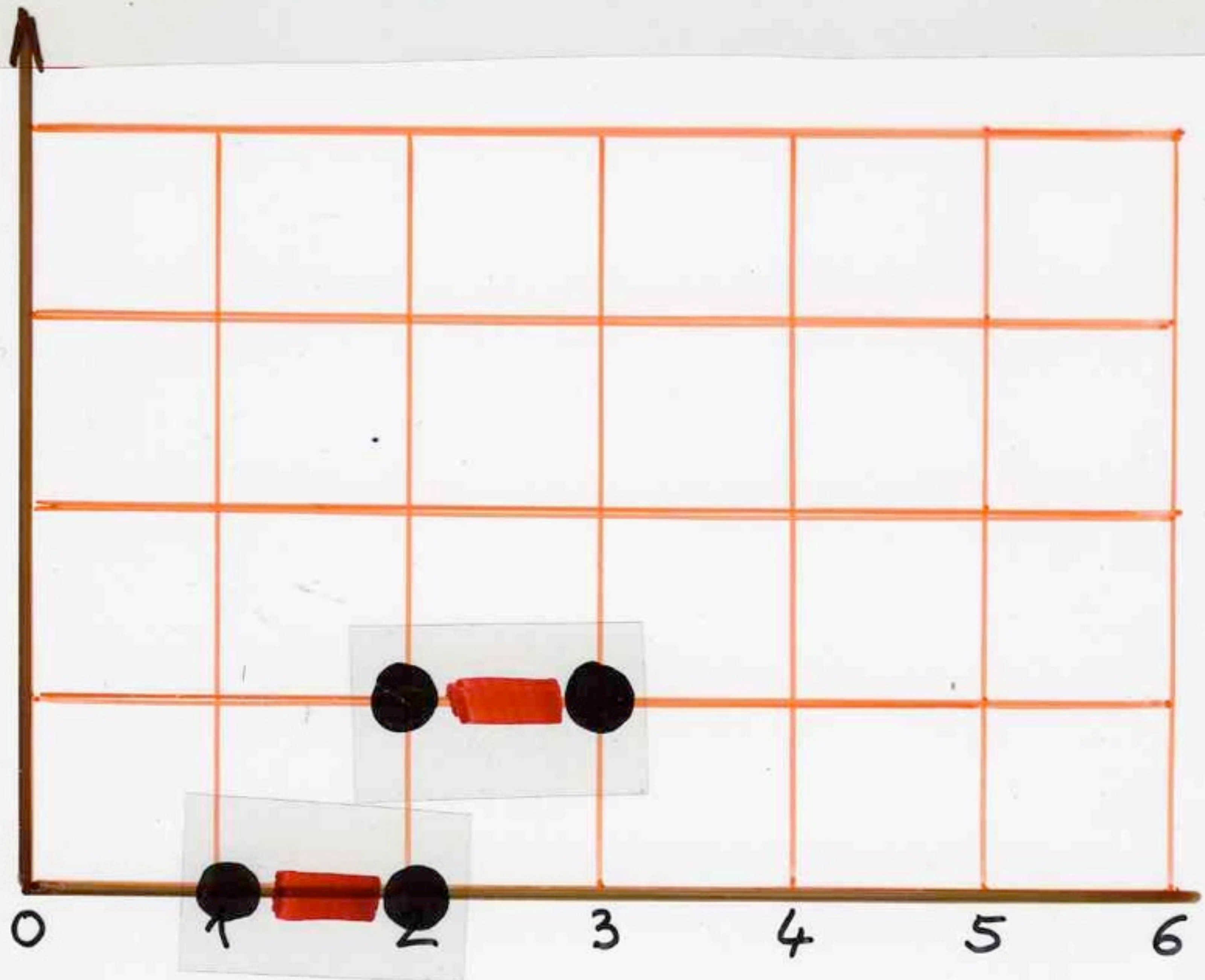
iff $|i-j| \geq 2$

$$w = \sigma_1 \sigma_2 \sigma_4 \sigma_1 \sigma_4 \sigma_3 \sigma_0 \sigma_4$$

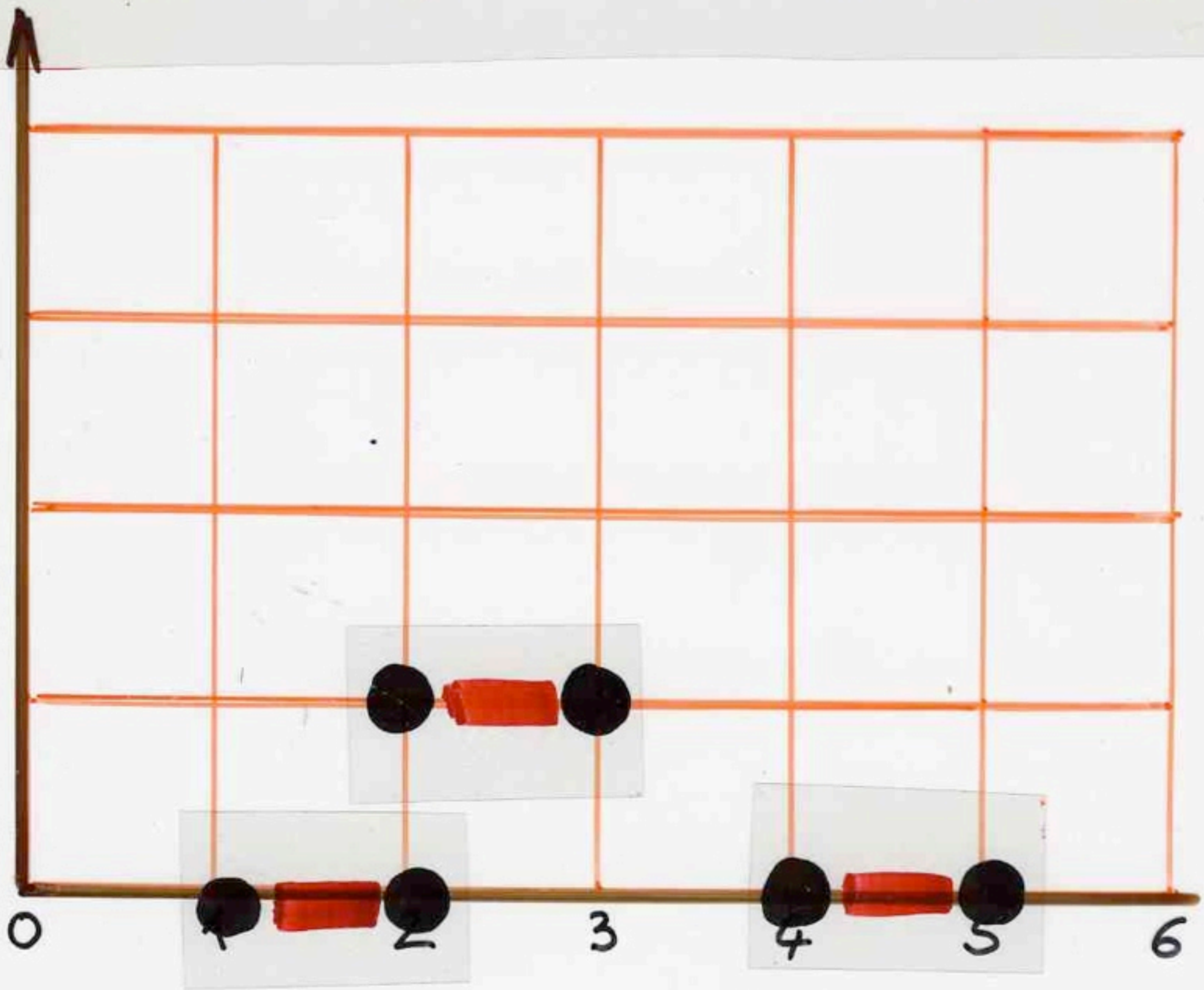
$$w = \sigma_1 \sigma_2 \sigma_4 \sigma_1 \sigma_4 \sigma_3 \sigma_0 \sigma_4$$



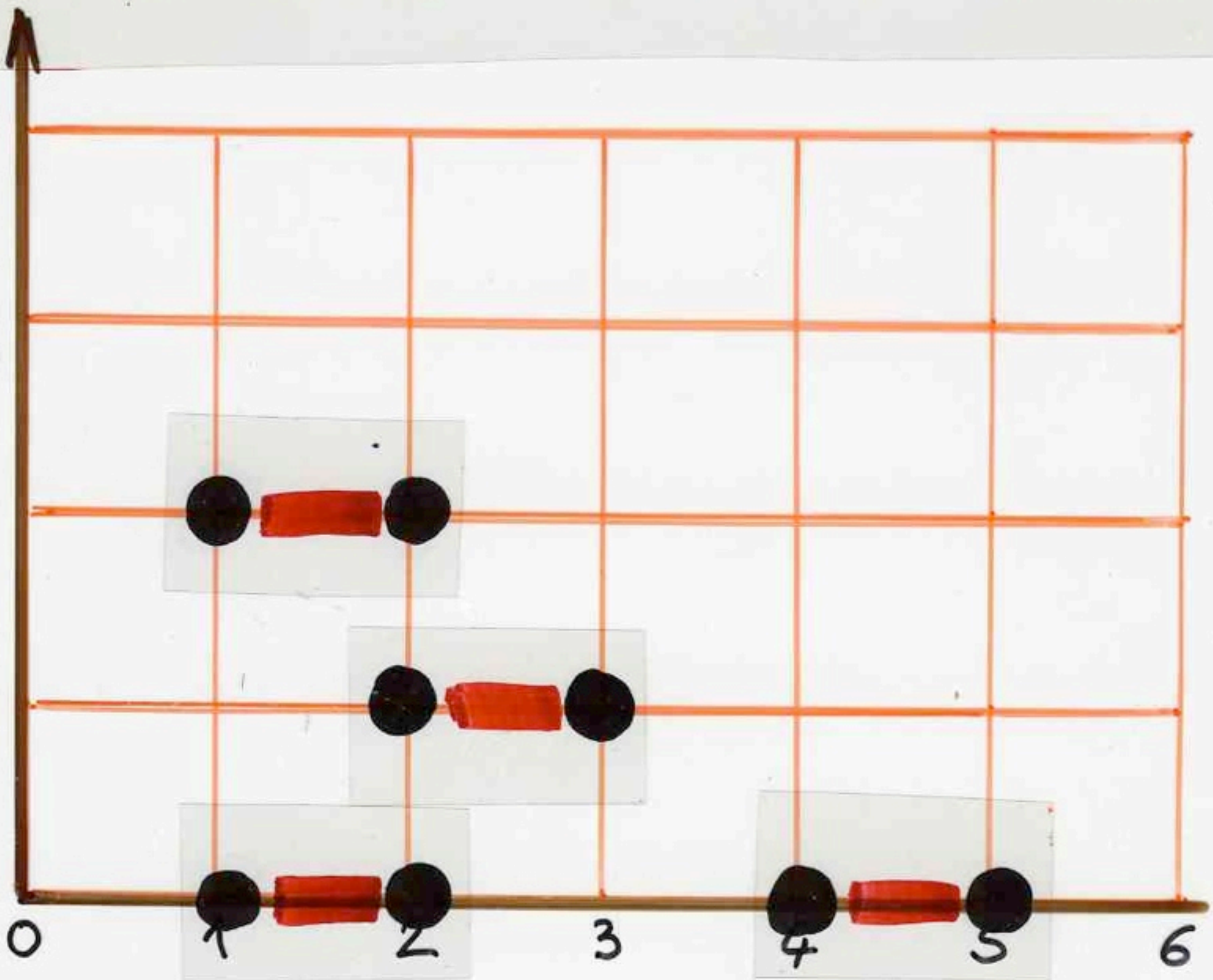
$$w = \sigma_1 \sigma_2 \sigma_4 \sigma_1 \sigma_4 \sigma_3 \sigma_0 \sigma_4$$



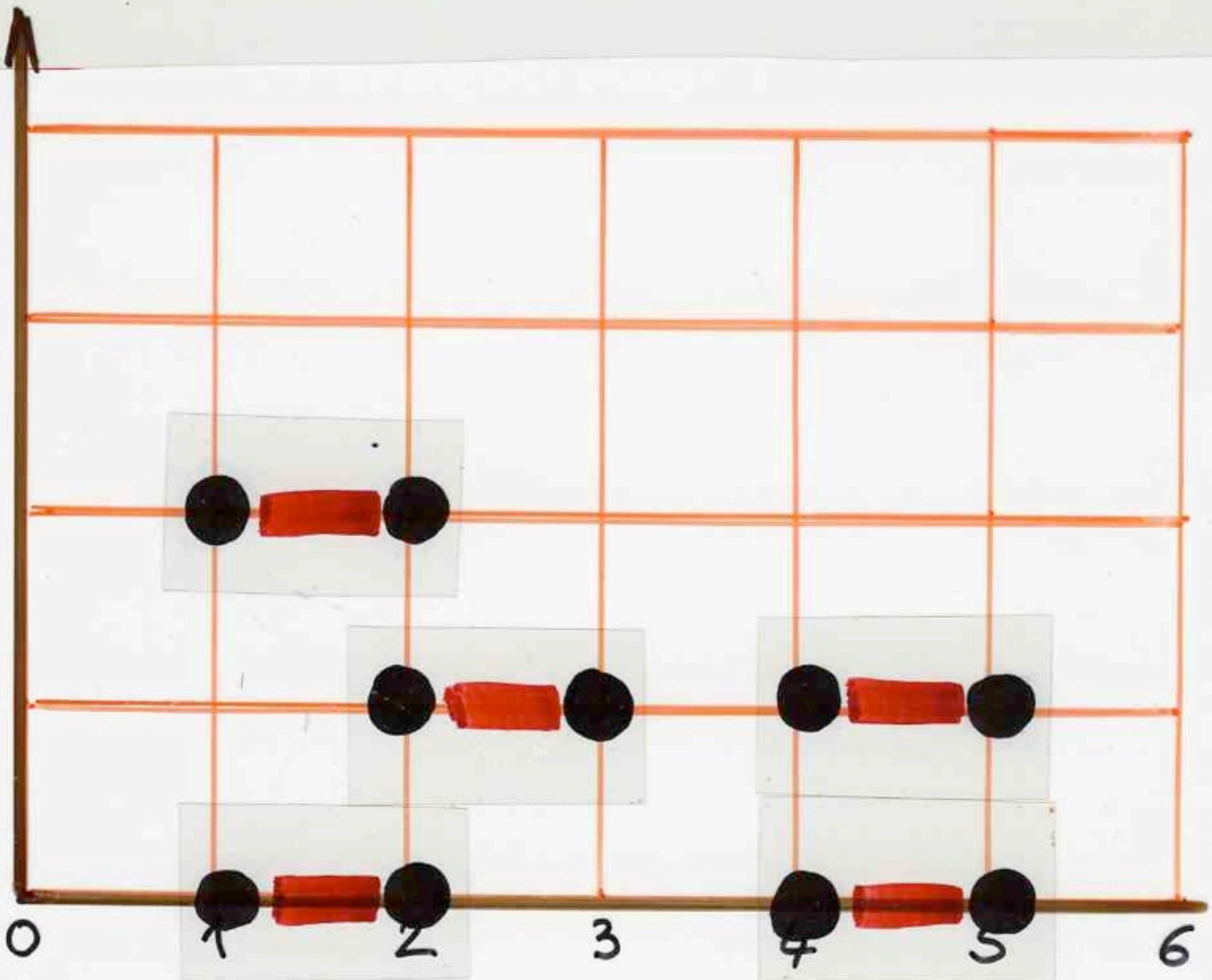
$$w = \sigma_1 \sigma_2 \sigma_4 \sigma_1 \sigma_4 \sigma_3 \sigma_0 \sigma_4$$



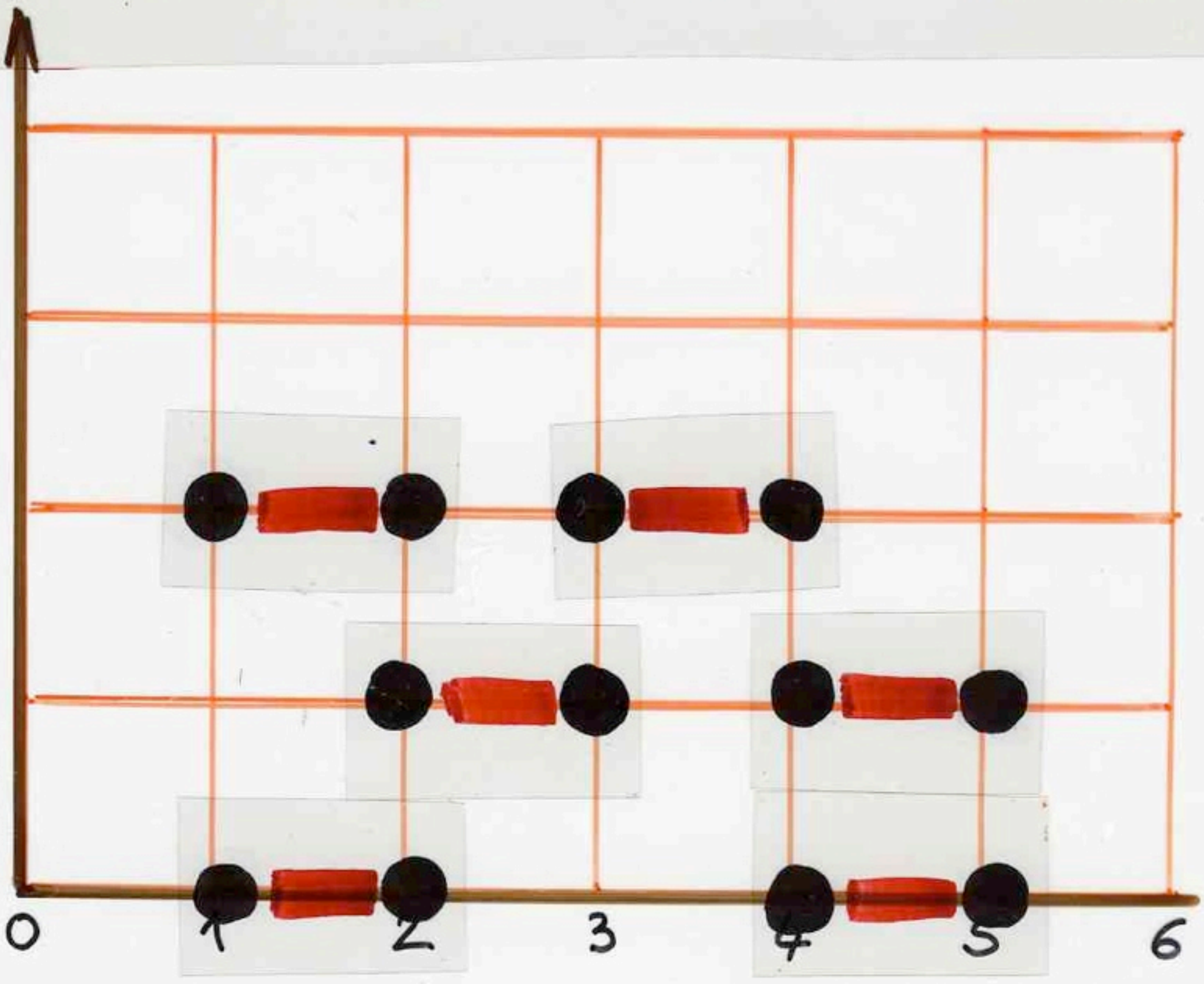
$$w = \sigma_1 \sigma_2 \sigma_4 \sigma_1 \sigma_4 \sigma_3 \sigma_0 \sigma_4$$



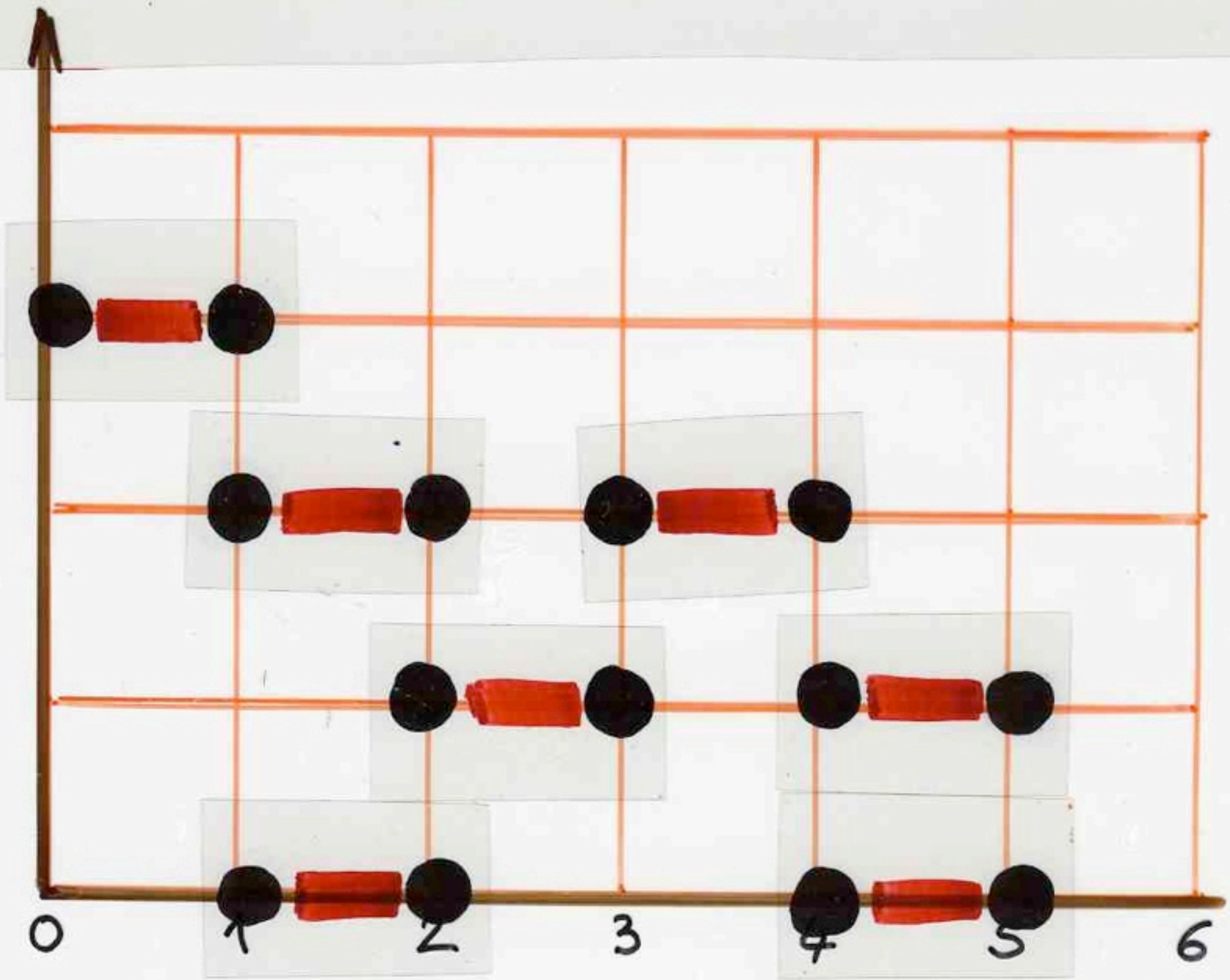
$$w = \sigma_1 \sigma_2 \sigma_4 \sigma_1 \sigma_4 \sigma_3 \sigma_0 \sigma_4$$



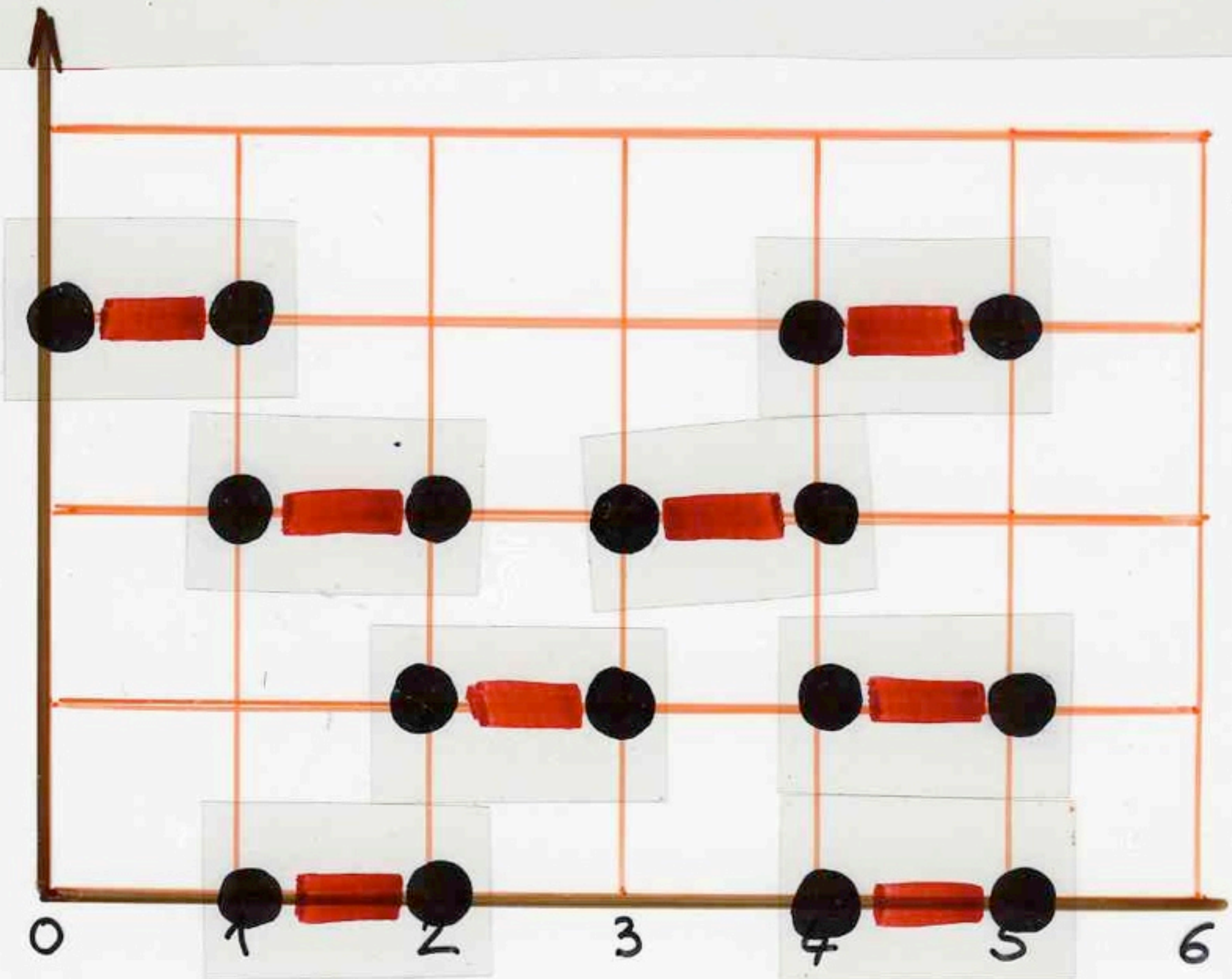
$$w = \sigma_1 \sigma_2 \sigma_4 \sigma_1 \sigma_4 \sigma_3 \sigma_0 \sigma_4$$



$$w = \sigma_1 \sigma_2 \sigma_4 \sigma_1 \sigma_4 \sigma_3 \sigma_0 \sigma_4$$

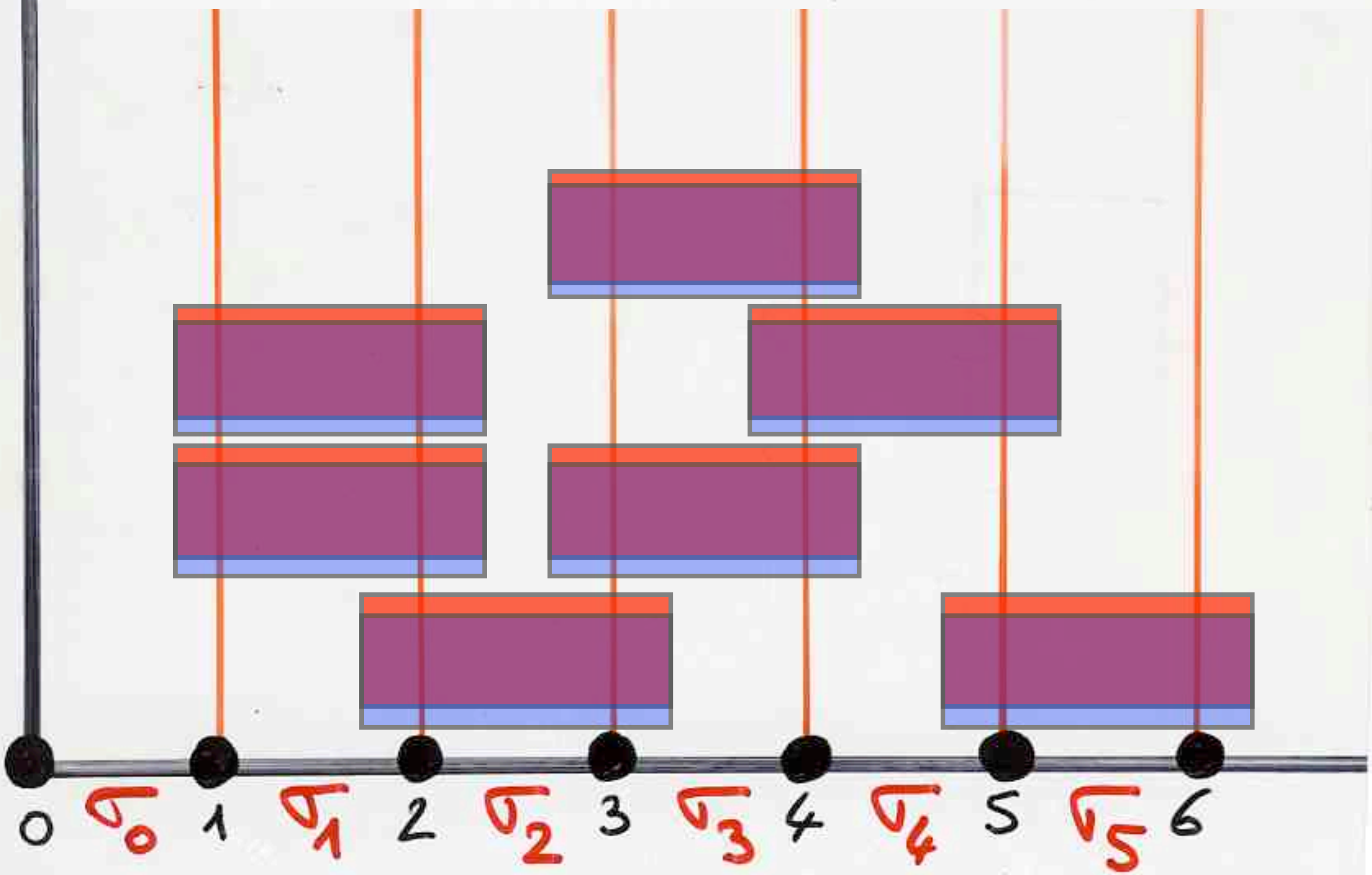


$$w = \sigma_1 \sigma_2 \sigma_4 \sigma_1 \sigma_4 \sigma_3 \sigma_0 \sigma_4$$



$$W = \sigma_2 \, \sigma_3 \, \sigma_5 \, \sigma_1 \, \sigma_4 \, \sigma_1 \, \sigma_3$$

$$W = \sigma_5 \, \sigma_2 \, \sigma_1 \, \sigma_1 \, \sigma_3 \, \sigma_4 \, \sigma_3$$



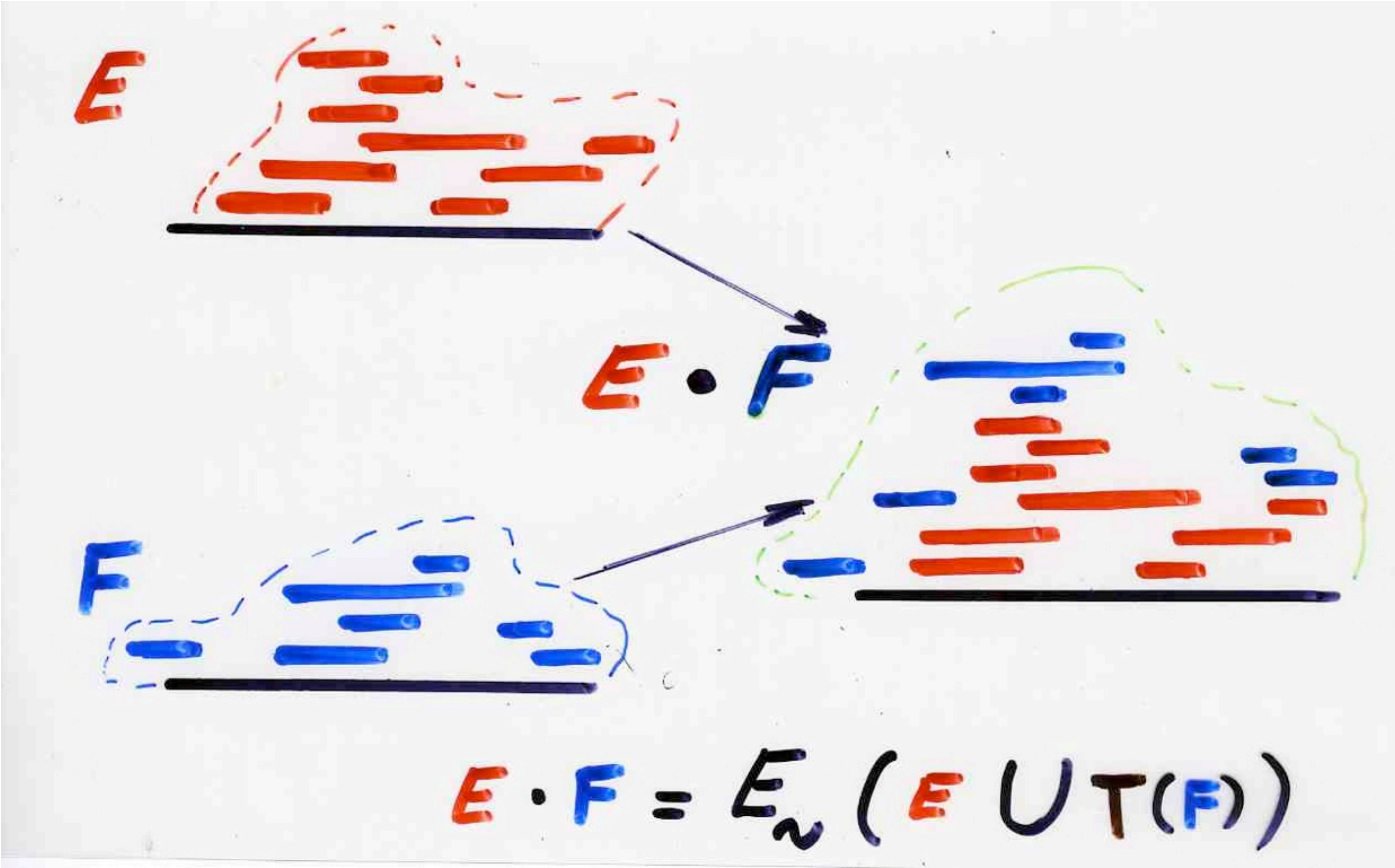
Proposition

$$\mathrm{Heap}(P, \mathcal{C}) \simeq P^{*} / \equiv_{C}$$

commutation monoid

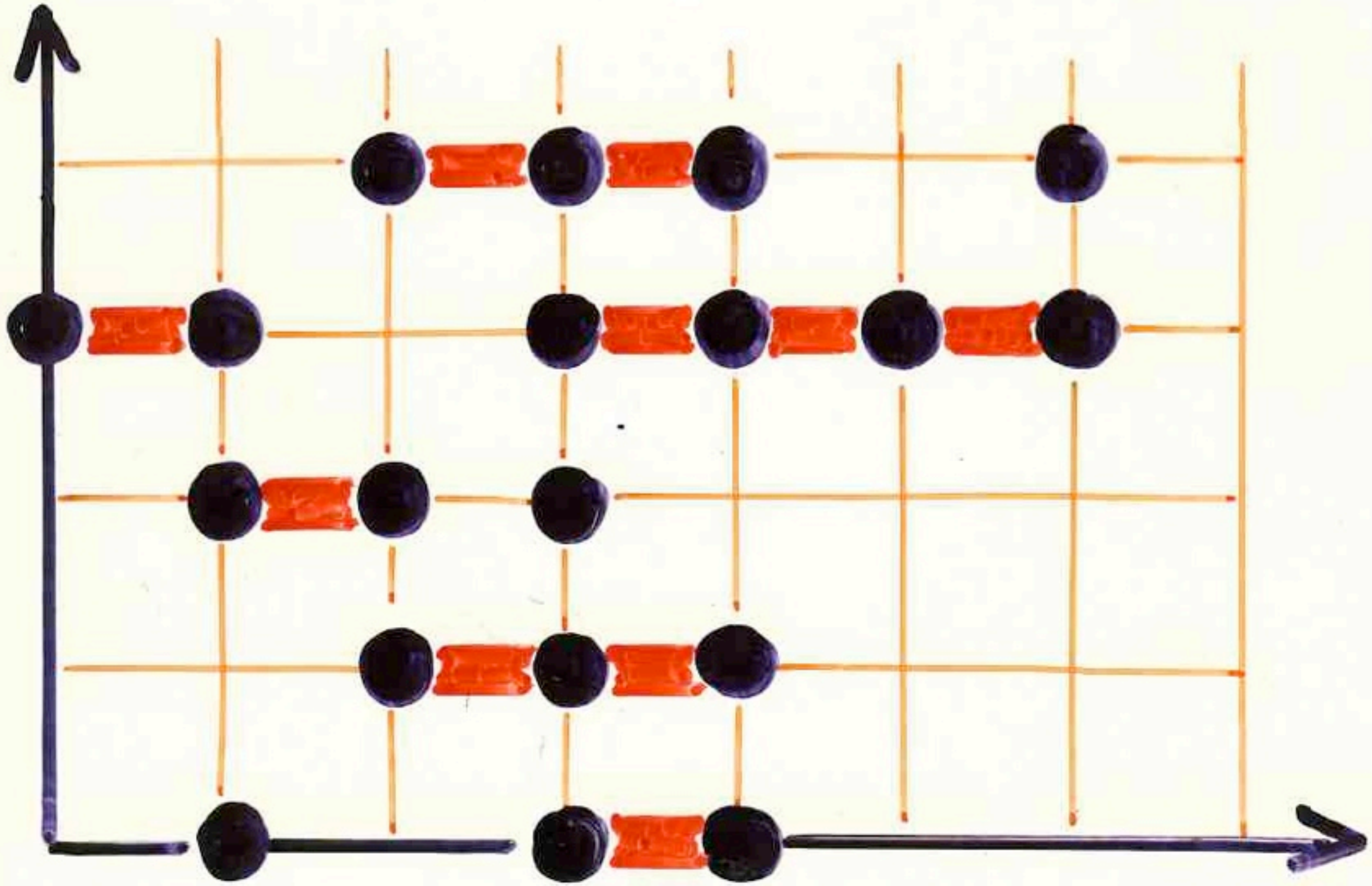
$$C = \overline{\mathcal{C}}$$

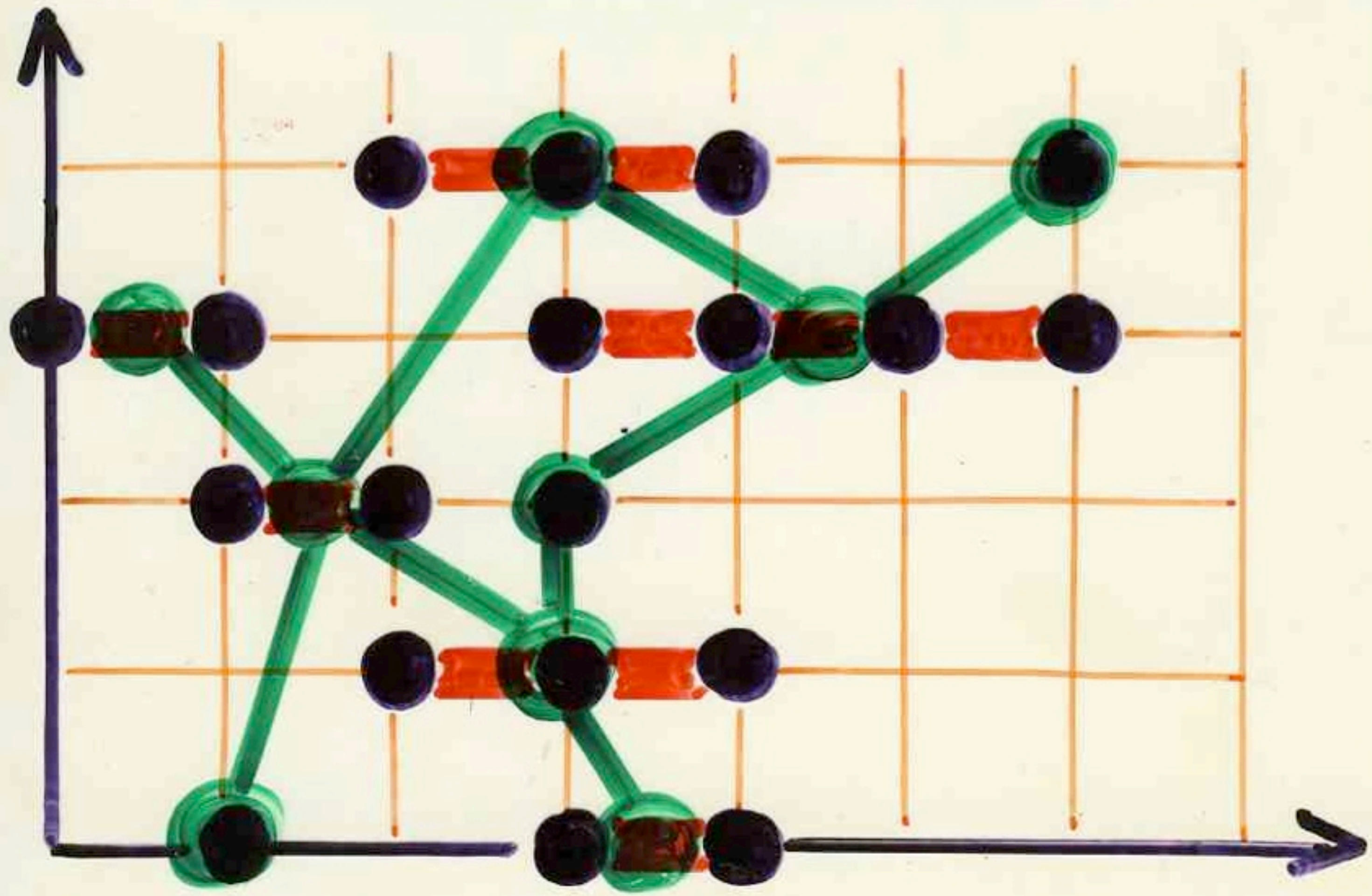
complementary relation

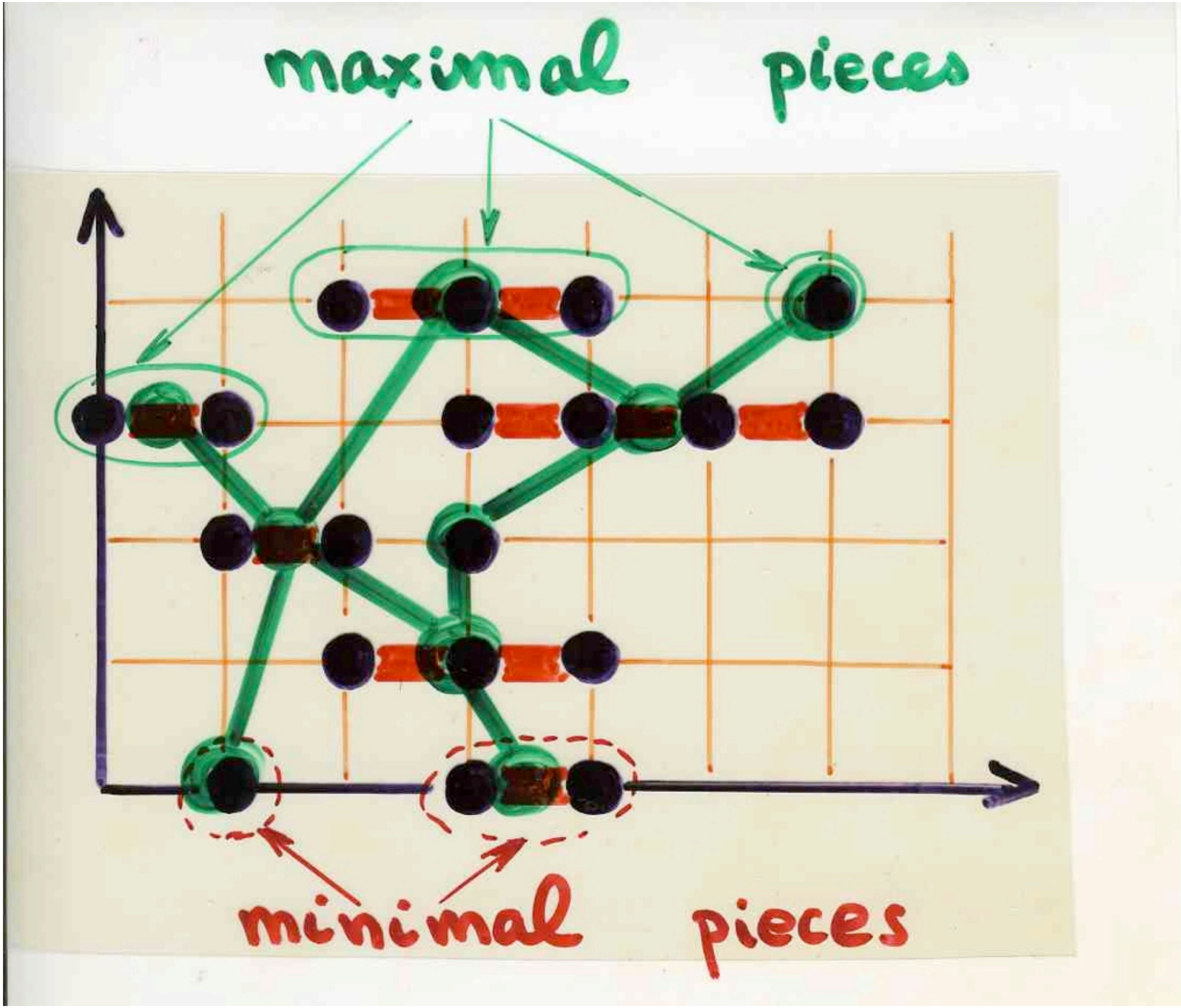


heaps monoid

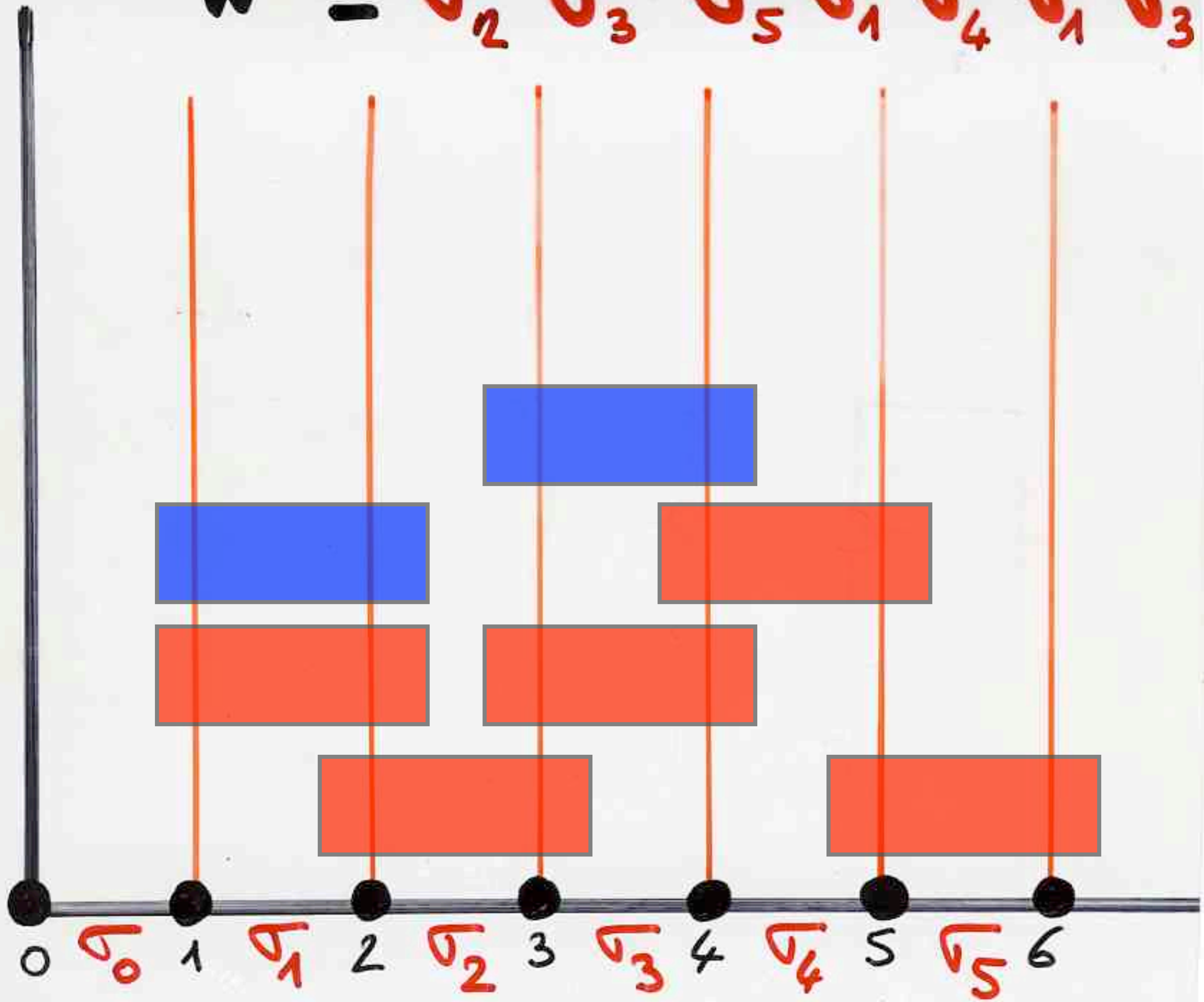
§3 Heaps and posets






maximal pieces
minimal pieces

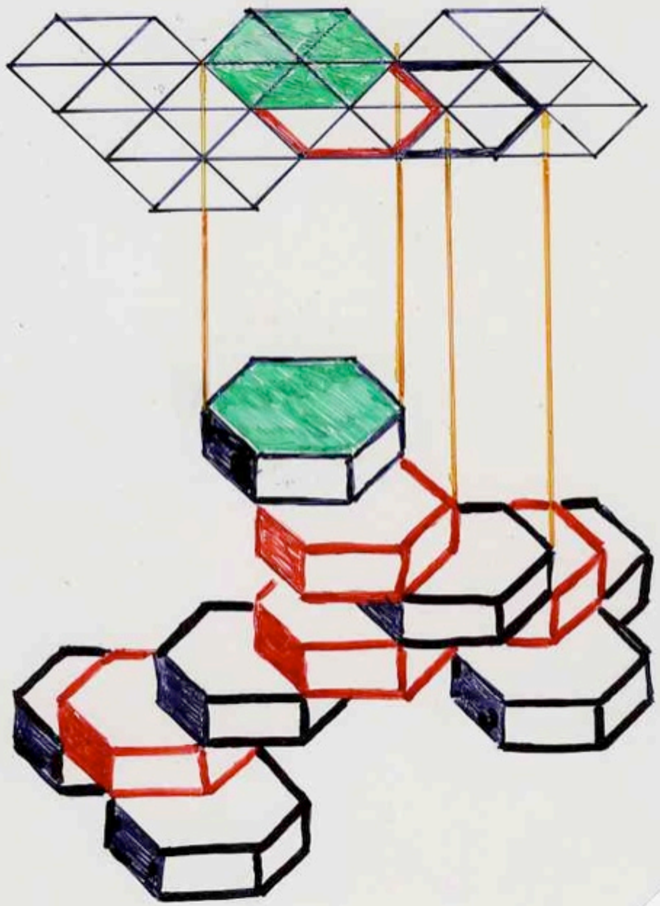
$$w = \sigma_2 \sigma_3 \sigma_5 \sigma_1 \sigma_4 \sigma_1 \sigma_3$$



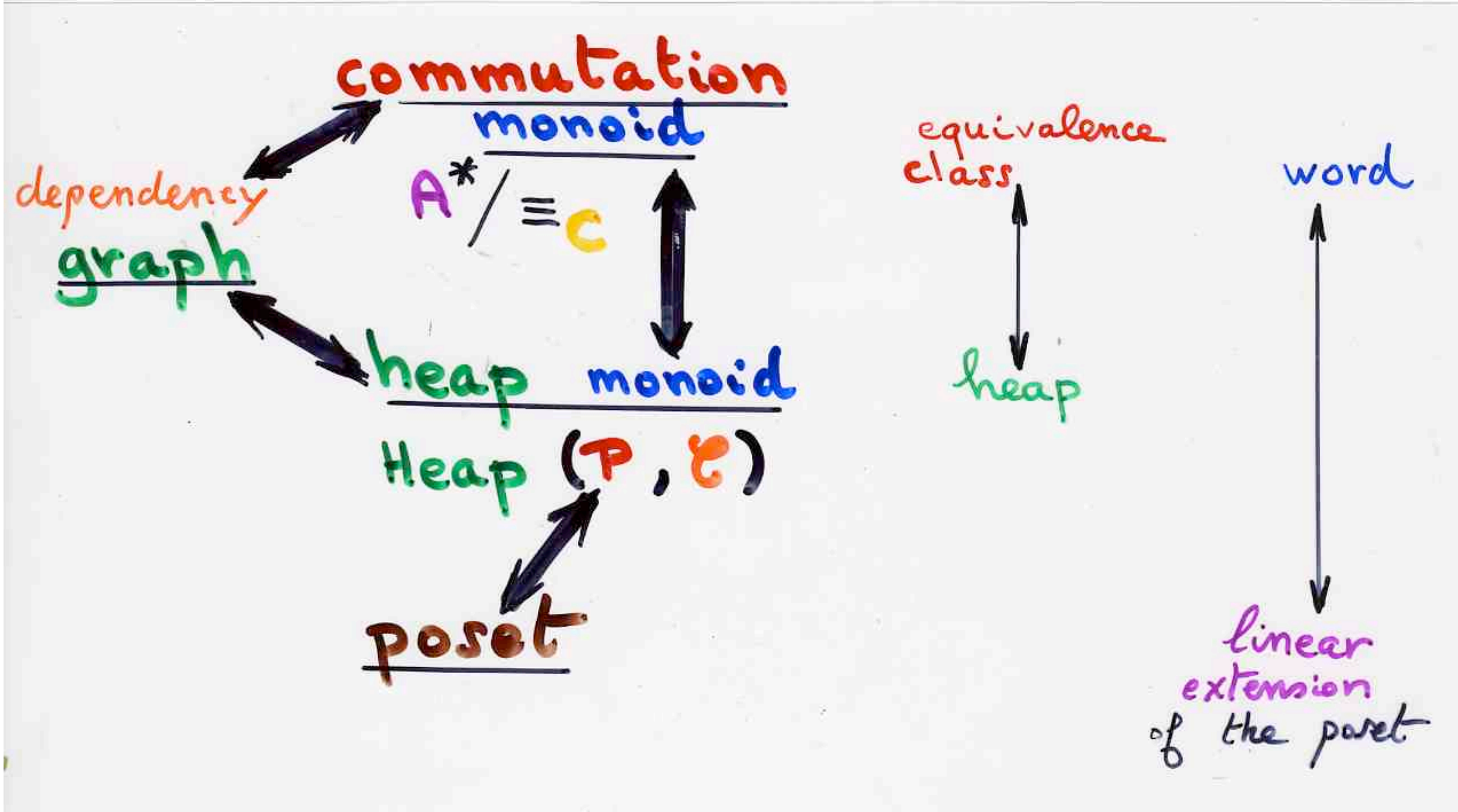
Pyramid

Def- Heap having only one maximal piece

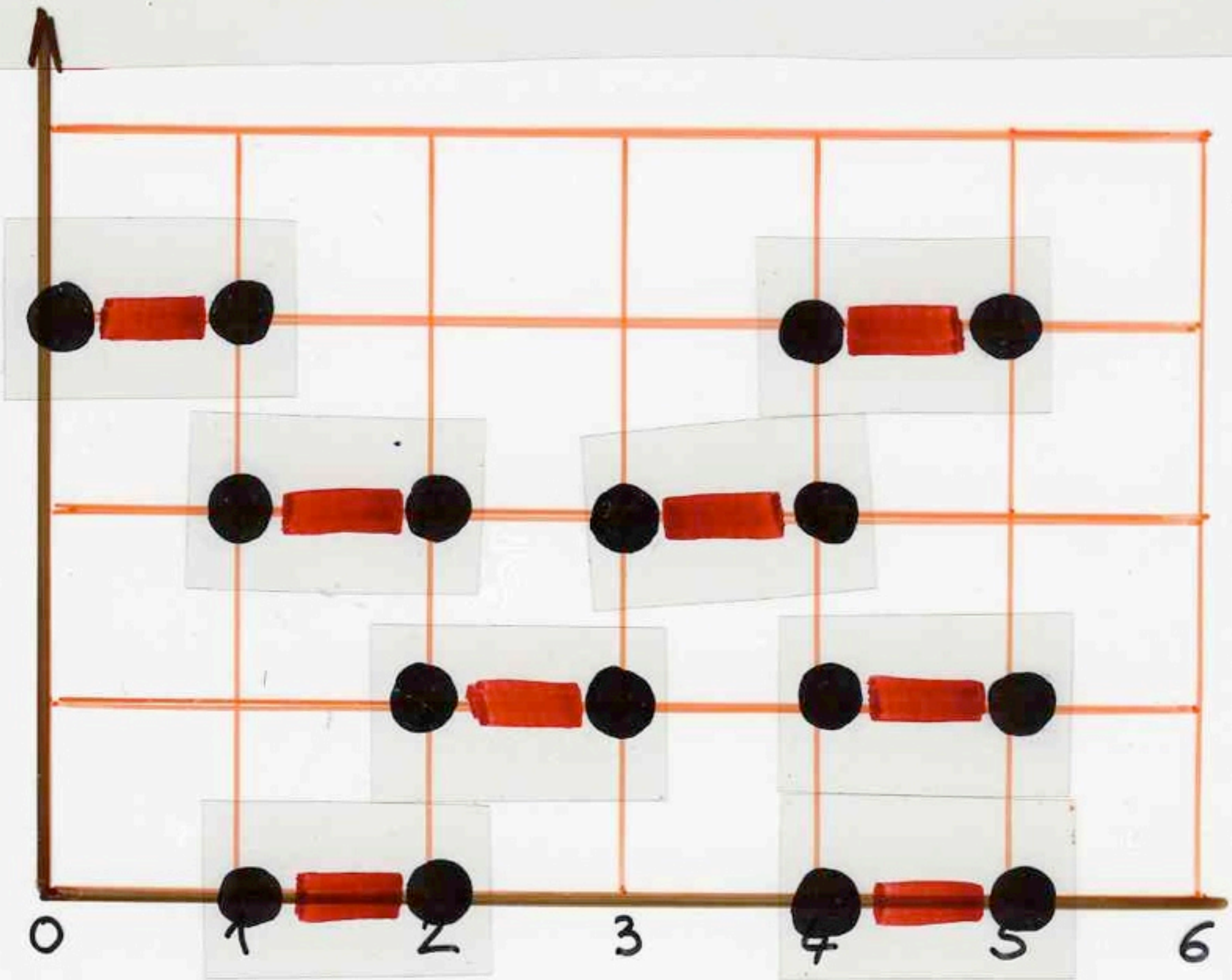
$$-p(-t) = y$$



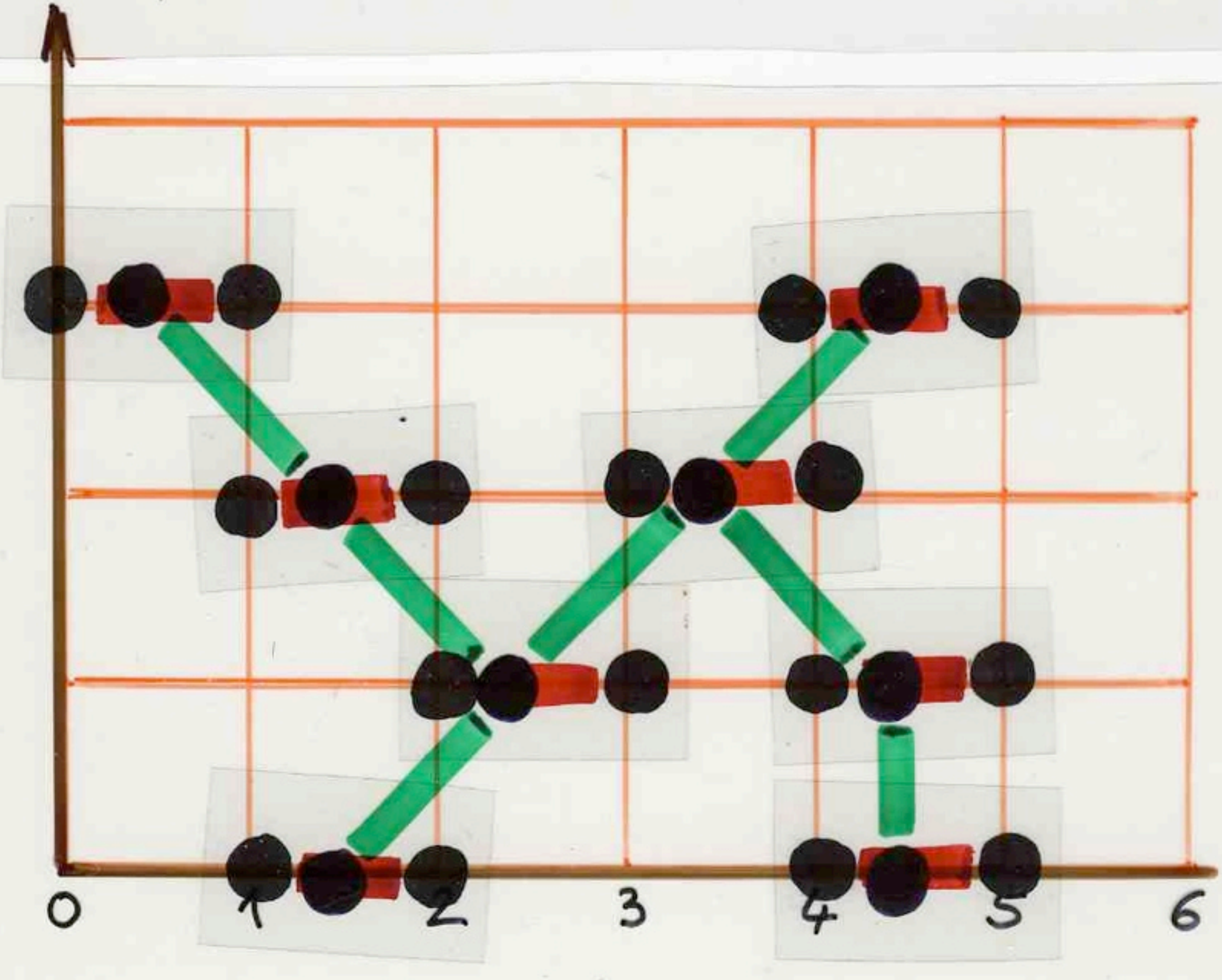
KC.


commutation
monoid
$A^*/\equiv_C$
dependency
graph
heap monoid
Heap (P, ξ)
poset
equivalence
class
heap
word
linear
extension
of the poset

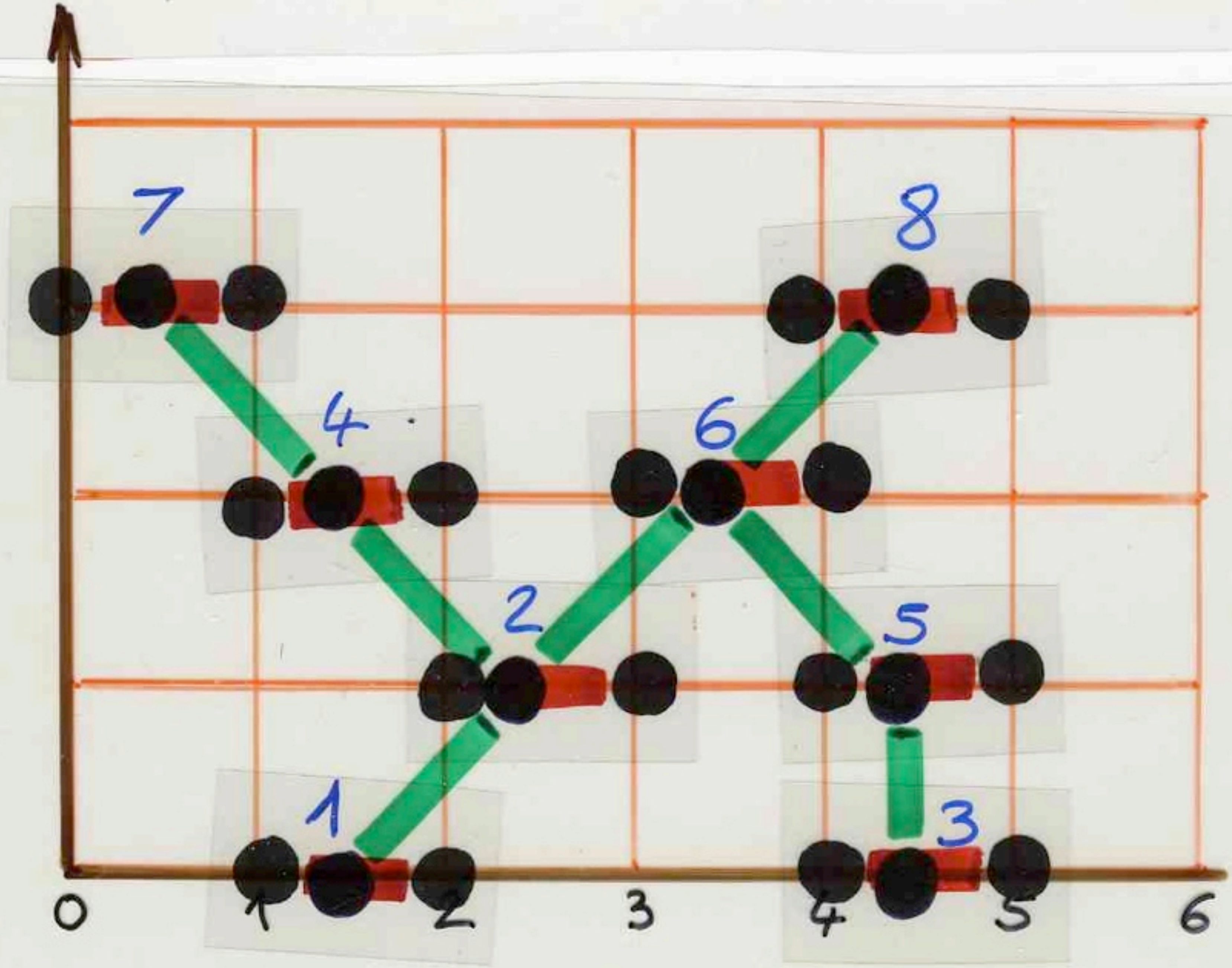
$$w = \sigma_1 \sigma_2 \sigma_4 \sigma_1 \sigma_4 \sigma_3 \sigma_0 \sigma_4$$

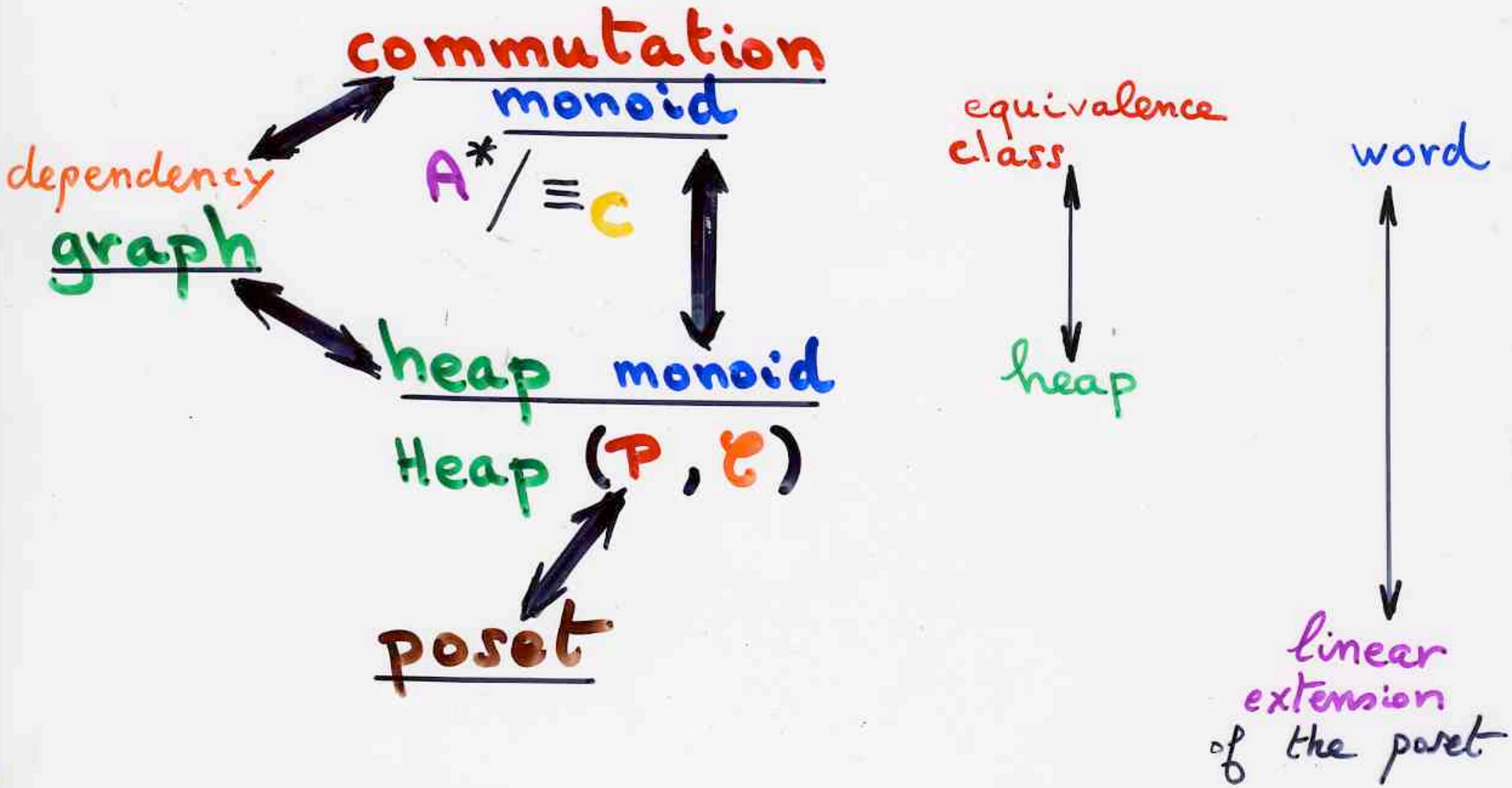


$$w = \sigma_1 \sigma_2 \sigma_4 \sigma_1 \sigma_4 \sigma_3 \sigma_0 \sigma_4$$



$$w = \sigma_1 \sigma_2 \sigma_4 \sigma_1 \sigma_4 \sigma_3 \sigma_0 \sigma_4$$




commutation monoid
$A^*/\equiv_C$
dependency graph
heap monoid
Heap (P, ζ)
poset
equivalence class
heap
word
linear extension of the poset