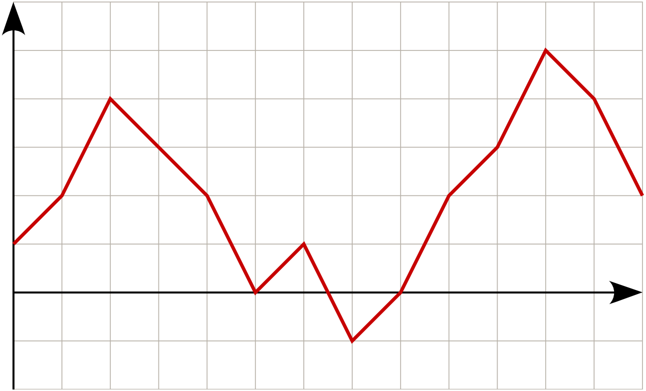
Le lien entre Michael Jordan et Catalan

Jérémie Bettinelli

Éric Fusy Cécile Mailler Lucas Randazzo

March 20, 2017

Basketball walks



Basketball walk: integer-valued walk with step-set $\{-2, -1, +1, +2\}$

Generating functions

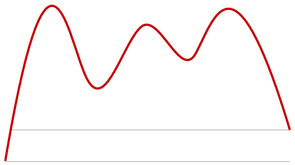
Theorem (Banderier & Krattenthaler & Krinik & Kruchinin & Kruchinin & Nguyen & Wallner '16)

The generating function G of basketball walks from 0 to 1 that are positive except at the origin, counted with weight z per step is given by

$$G(z) = -\frac{1}{2} + \frac{1}{2}\sqrt{\frac{2-3z-2\sqrt{1-4z}}{z}}.$$

$\mathcal{G} := \{$basketball walks from 0 to 1 that are positive except at the origin$\}$

$|\mathfrak{w}|$: number of steps of $\mathfrak{w}$



$$G(z) := \sum_{\mathfrak{w}\in\mathcal{G}} z^{|\mathfrak{w}|} = \sum_{n=0}^{\infty} \left|\{\mathfrak{w}\in\mathcal{G} : |\mathfrak{w}| = n\}\right| z^n$$

Catalan is everywhere!

The previous authors observed that

$$1 + G(z) + G^2(z) = \mathrm{Cat}(z) \tag{1}$$

where Cat is the Catalan generating function.

$\mathrm{Cat}(z) = \sum_{n=0}^{\infty} c_n z^n$ where $c_n := \frac{1}{n+1}\binom{2n}{n}$ is the n-th Catalan number

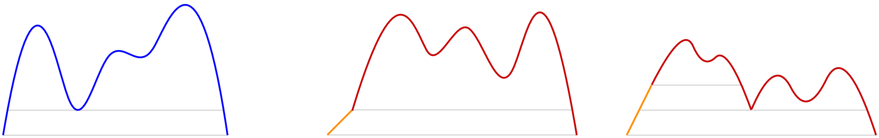
1, 1, 2, 5, 14, 42, 132, 429, 1430, 4862, 16796, 58786, 208012

n-edge rooted trees, $n+1$-leaf binary rooted trees, $2n$-step Dyck walks, well-parenthesized words with n pairs of parentheses, rooted triangulations of the $n+2$-gon, noncrossing partitions of the n-set, etc.

C-walks

C-walk: basketball walk from 0 to 0 that visits 1 and is positive except at the extremities

$$\mathcal{C} := \{C\text{-walks}\} \qquad C(z) := \sum_{\mathfrak{w}\in\mathcal{C}} z^{|\mathfrak{w}|}$$

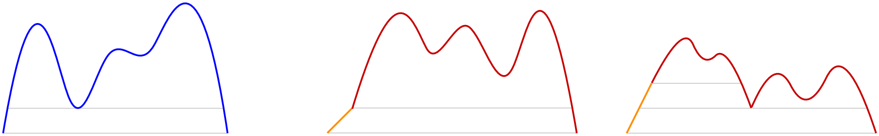


$$\mathcal{C} = \mathcal{Z}\,\mathcal{G} + \mathcal{Z}\,\mathcal{G}\,\mathcal{G}$$

C-walks

C-walk: basketball walk from 0 to 0 that visits 1 and is positive except at the extremities

$$\mathcal{C} := \{C\text{-walks}\} \qquad C(z) := \sum_{\mathfrak{w} \in \mathcal{C}} z^{|\mathfrak{w}|}$$



$$\mathcal{C} = \mathcal{Z}\ \mathcal{G} + \mathcal{Z}\ \mathcal{G}\ \mathcal{G}$$

$$C(z) = zG(z) + zG^2(z)$$

Equation (1) becomes $C(z) = z\left(\mathrm{Cat}(z) - 1\right)$, which is the generating function of nontrivial binary trees counted with weight z per leaf.

Refined enumeration

even step: step starting at even height
odd step: step starting at odd height

Proposition

The number of C-walks with $2d$ ± 1-steps, ℓ odd $+2$-steps or even -2-steps, and r odd -2-steps or even $+2$-steps is equal to

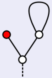
$$\frac{1}{d}\binom{2d-2}{d-1}\binom{\ell+r+2d-2}{\ell+r}\binom{\ell+r}{\ell}.$$

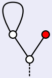
Matched statistics

Proposition

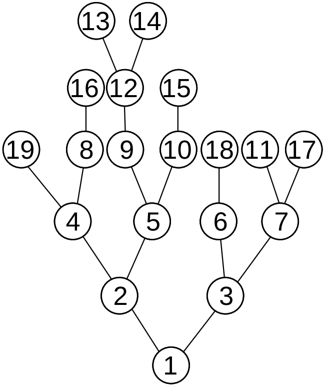
n-step *C*-walk $\longleftrightarrow$ *n*-leaf binary tree

± 1-steps  or  $\longleftrightarrow$  double leaves

odd $+2$ / *even* -2  or  $\longleftrightarrow$  left leaves

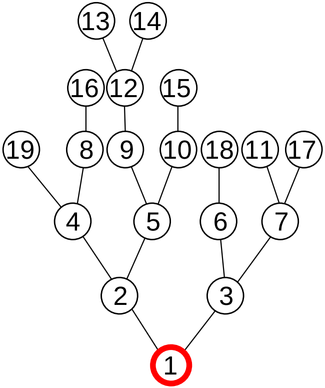
even $+2$ / *odd* -2  or  $\longleftrightarrow$  right leaves

Increasing unary-binary tree



increasing unary-binary tree of size n: plane tree with n vertices labeled 1, 2, . . . , n such that each vertex has at most 2 children, and all have larger labels

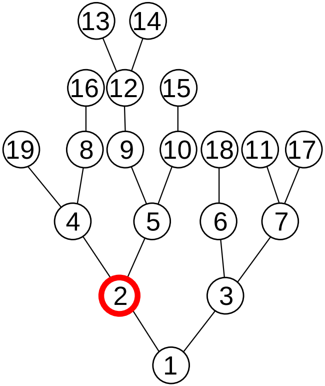
Increasing unary-binary tree

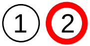


increasing unary-binary tree of size n: plane tree with n vertices labeled 1, 2, . . . , n such that each vertex has at most 2 children, and all have larger labels

We associate with it the permutation obtained by reading the labels of the tree in breadth-first search order

Increasing unary-binary tree

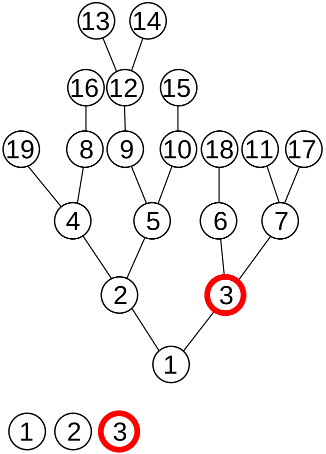




increasing unary-binary tree of size n: plane tree with n vertices labeled 1, 2, …, n such that each vertex has at most 2 children, and all have larger labels

We associate with it the permutation obtained by reading the labels of the tree in breadth-first search order

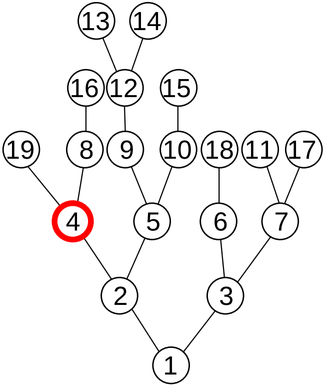
Increasing unary-binary tree

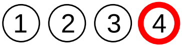


increasing unary-binary tree of size n: plane tree with n vertices labeled 1, 2, . . . , n such that each vertex has at most 2 children, and all have larger labels

We associate with it the permutation obtained by reading the labels of the tree in breadth-first search order

Increasing unary-binary tree

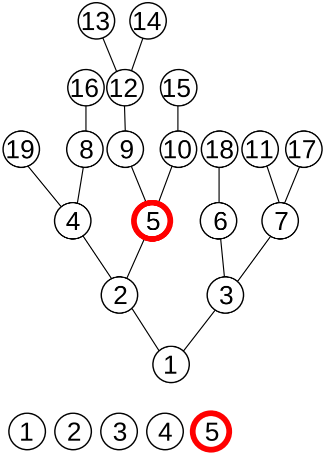




increasing unary-binary tree of size *n*: plane tree with *n* vertices labeled 1, 2, . . . , *n* such that each vertex has at most 2 children, and all have larger labels

We associate with it the permutation obtained by reading the labels of the tree in breadth-first search order

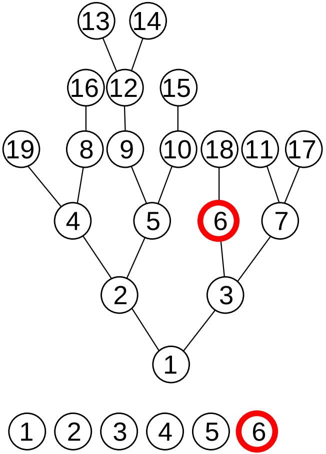
Increasing unary-binary tree



increasing unary-binary tree of size n: plane tree with n vertices labeled 1, 2, ..., n such that each vertex has at most 2 children, and all have larger labels

We associate with it the permutation obtained by reading the labels of the tree in breadth-first search order

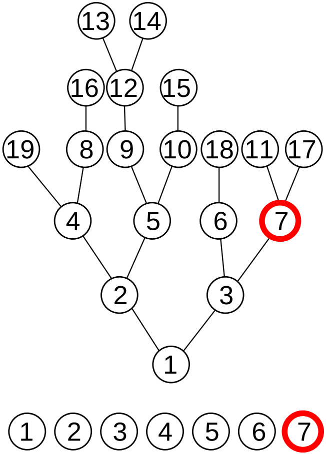
Increasing unary-binary tree



increasing unary-binary tree of size n: plane tree with n vertices labeled 1, 2, ..., n such that each vertex has at most 2 children, and all have larger labels

We associate with it the permutation obtained by reading the labels of the tree in breadth-first search order

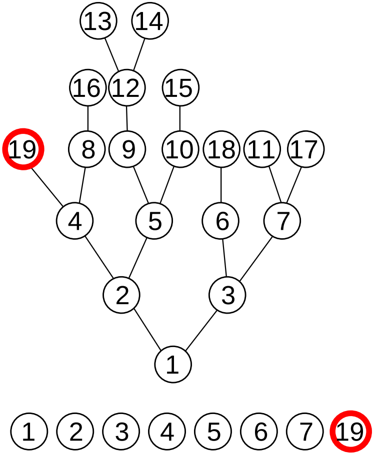
Increasing unary-binary tree



increasing unary-binary tree of size n: plane tree with n vertices labeled 1, 2, . . . , n such that each vertex has at most 2 children, and all have larger labels

We associate with it the permutation obtained by reading the labels of the tree in breadth-first search order

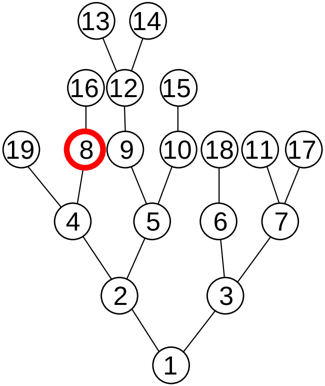
Increasing unary-binary tree



increasing unary-binary tree of size *n*: plane tree with *n* vertices labeled 1, 2, . . . , *n* such that each vertex has at most 2 children, and all have larger labels

We associate with it the permutation obtained by reading the labels of the tree in breadth-first search order

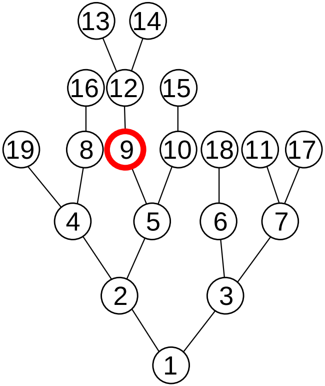
Increasing unary-binary tree



increasing unary-binary tree of size *n*: plane tree with *n* vertices labeled 1, 2, …, *n* such that each vertex has at most 2 children, and all have larger labels

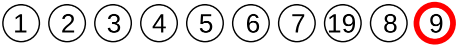
We associate with it the permutation obtained by reading the labels of the tree in breadth-first search order

Increasing unary-binary tree

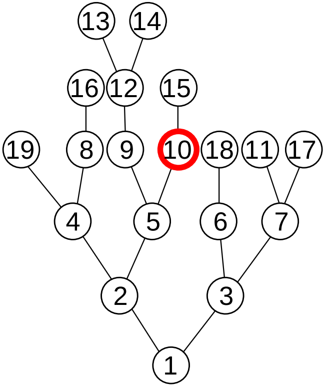


increasing unary-binary tree of size *n*: plane tree with *n* vertices labeled 1, 2, . . . , *n* such that each vertex has at most 2 children, and all have larger labels

We associate with it the permutation obtained by reading the labels of the tree in breadth-first search order



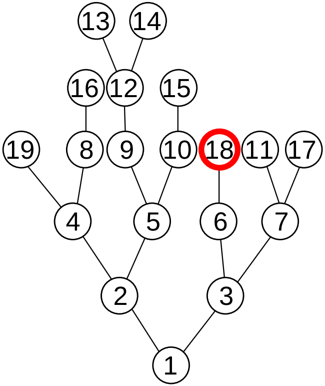
Increasing unary-binary tree



increasing unary-binary tree of size *n*: plane tree with *n* vertices labeled 1, 2, ..., *n* such that each vertex has at most 2 children, and all have larger labels

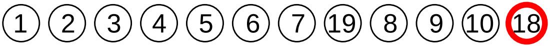
We associate with it the permutation obtained by reading the labels of the tree in breadth-first search order

Increasing unary-binary tree

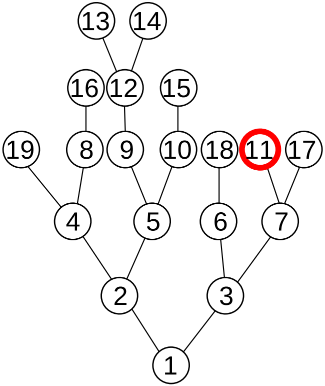


increasing unary-binary tree of size n: plane tree with n vertices labeled 1, 2, . . . , n such that each vertex has at most 2 children, and all have larger labels

We associate with it the permutation obtained by reading the labels of the tree in breadth-first search order



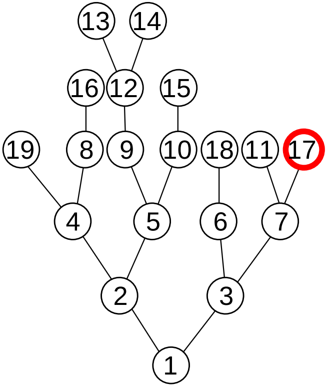
Increasing unary-binary tree



increasing unary-binary tree of size n: plane tree with n vertices labeled 1, 2, . . . , n such that each vertex has at most 2 children, and all have larger labels

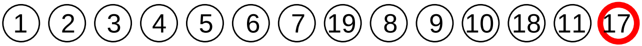
We associate with it the permutation obtained by reading the labels of the tree in breadth-first search order

Increasing unary-binary tree

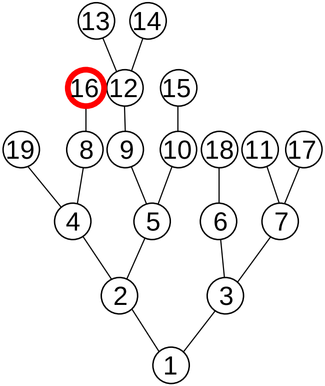


increasing unary-binary tree of size n: plane tree with n vertices labeled 1, 2, …, n such that each vertex has at most 2 children, and all have larger labels

We associate with it the permutation obtained by reading the labels of the tree in breadth-first search order



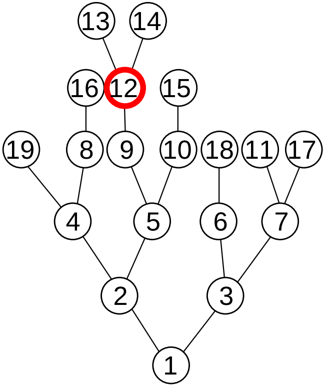
Increasing unary-binary tree



increasing unary-binary tree of size *n*: plane tree with *n* vertices labeled 1, 2, . . . , *n* such that each vertex has at most 2 children, and all have larger labels

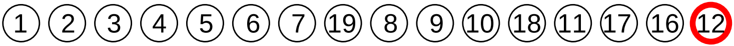
We associate with it the permutation obtained by reading the labels of the tree in breadth-first search order

Increasing unary-binary tree

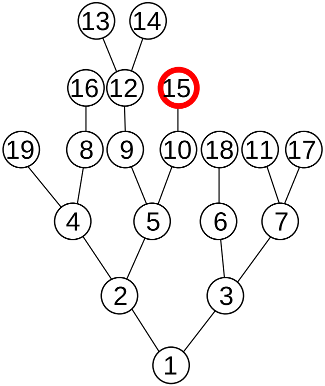


increasing unary-binary tree of size n: plane tree with n vertices labeled 1, 2, ..., n such that each vertex has at most 2 children, and all have larger labels

We associate with it the permutation obtained by reading the labels of the tree in breadth-first search order



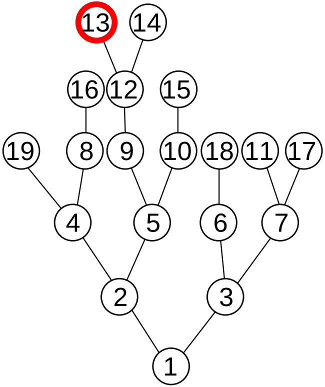
Increasing unary-binary tree



increasing unary-binary tree of size n: plane tree with n vertices labeled 1, 2, . . . , n such that each vertex has at most 2 children, and all have larger labels

We associate with it the permutation obtained by reading the labels of the tree in breadth-first search order

Increasing unary-binary tree

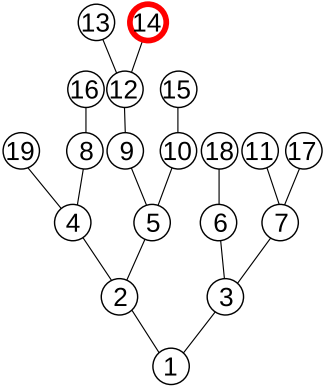


increasing unary-binary tree of size n: plane tree with n vertices labeled 1, 2, …, n such that each vertex has at most 2 children, and all have larger labels

We associate with it the permutation obtained by reading the labels of the tree in breadth-first search order

1 2 3 4 5 6 7 19 8 9 10 18 11 17 16 12 15 13

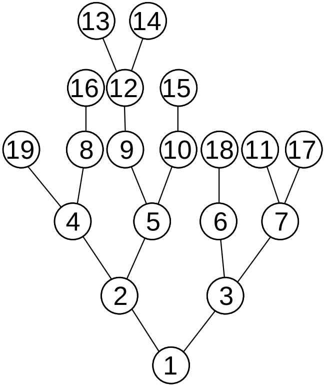
Increasing unary-binary tree



increasing unary-binary tree of size n: plane tree with n vertices labeled 1, 2, . . . , n such that each vertex has at most 2 children, and all have larger labels

We associate with it the permutation obtained by reading the labels of the tree in breadth-first search order

Increasing unary-binary tree

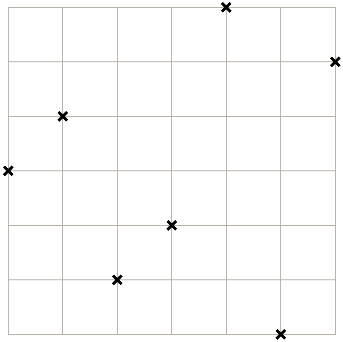


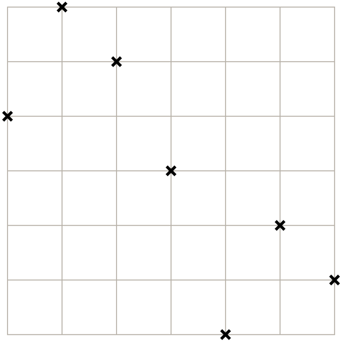
increasing unary-binary tree of size n: plane tree with n vertices labeled 1, 2, . . . , n such that each vertex has at most 2 children, and all have larger labels

We associate with it the permutation obtained by reading the labels of the tree in breadth-first search order

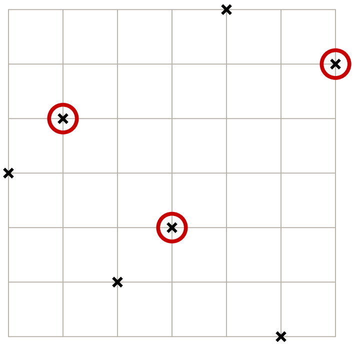
1 2 3 4 5 6 7 19 8 9 10 18 11 17 16 12 15 13 14

Permutation avoiding 213

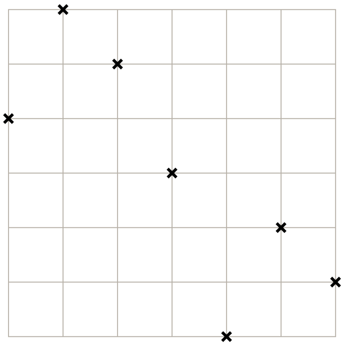




Permutation avoiding 213



contains 213



avoids 213

Counting IUBTs

IUBT: increasing unary-binary tree with associated permutation avoiding 213

Theorem

IUBTs are counted by G-walks (basketball walks from 0 to 1 that are positive except at the origin).

Proposition

For $n \geq 1$ and $0 \leq k \leq \lfloor (n-1)/2 \rfloor$, the number of n-vertex IUBTs with exactly $n-1-2k$ unary nodes is

$$\frac{1}{n}\binom{2n}{k}\binom{n-k}{k+1}.$$

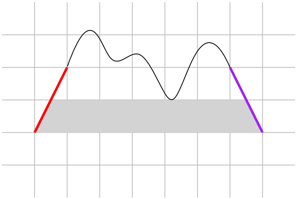
Matched statistics

IUBT: increasing unary-binary tree with associated permutation avoiding 213

Proposition

n-step G-walk $\longleftrightarrow$ *n-vertex IUBT*

staggered ± 2*-steps* $\longleftrightarrow$  *unary nodes*



The red and purple steps are paired.

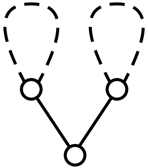
A staggered ± 2-step is a ± 2-step that is not paired with any other ± 2-step.

Decomposition of binary trees

$\mathcal{N}$: class of nontrivial binary trees counted by number of leaves

Goal

Understand bijectively that $\mathcal{C} = \mathcal{N}$



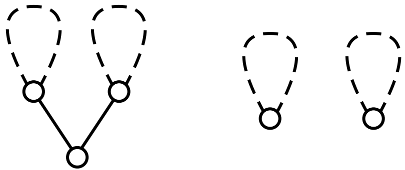
$\mathcal{N}$

Decomposition of binary trees

$\mathcal{N}$: class of nontrivial binary trees counted by number of leaves

Goal

Understand bijectively that $\mathcal{C} = \mathcal{N}$



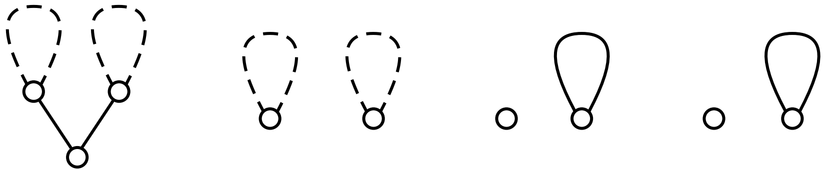
$\mathcal{N} \quad = \quad \times$

Decomposition of binary trees

$\mathcal{N}$: class of nontrivial binary trees counted by number of leaves

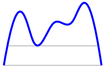
Goal

Understand bijectively that $\mathcal{C} = \mathcal{N}$



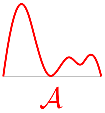
$$\mathcal{N} \quad = \quad \times \quad = (\mathcal{Z} + \mathcal{N}) \times (\mathcal{Z} + \mathcal{N})$$

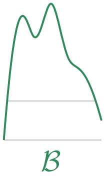
Elementary decomposition of basketball walks

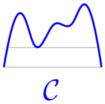


$\mathcal{C}$

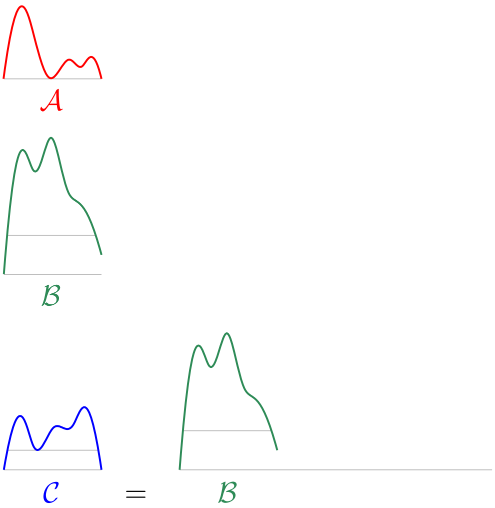
Elementary decomposition of basketball walks



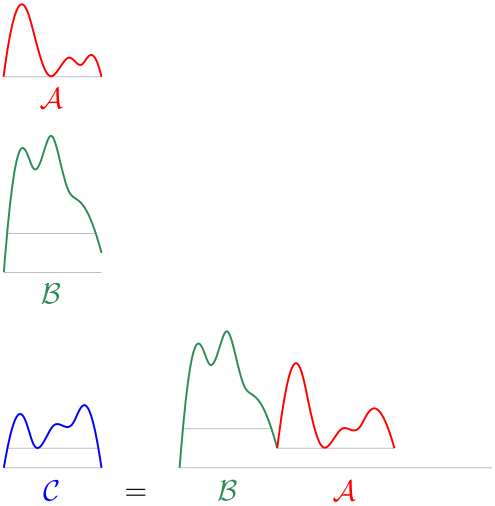




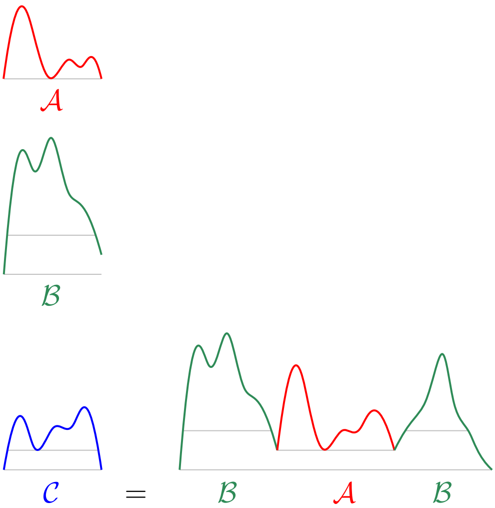
Elementary decomposition of basketball walks



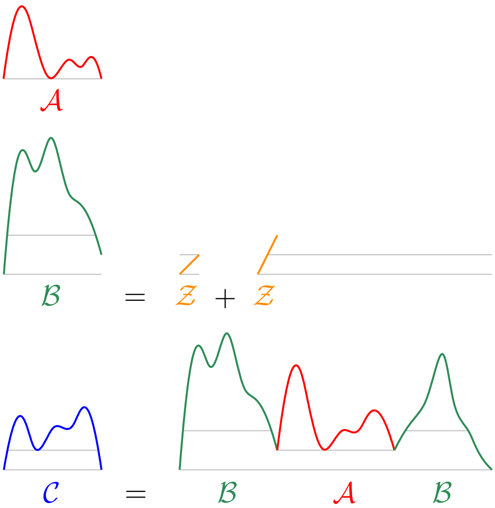
Elementary decomposition of basketball walks



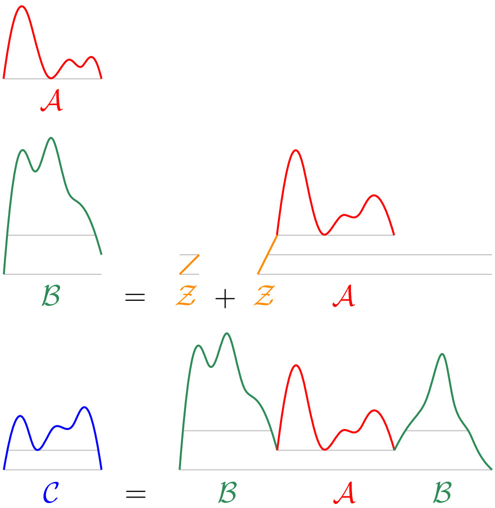
Elementary decomposition of basketball walks



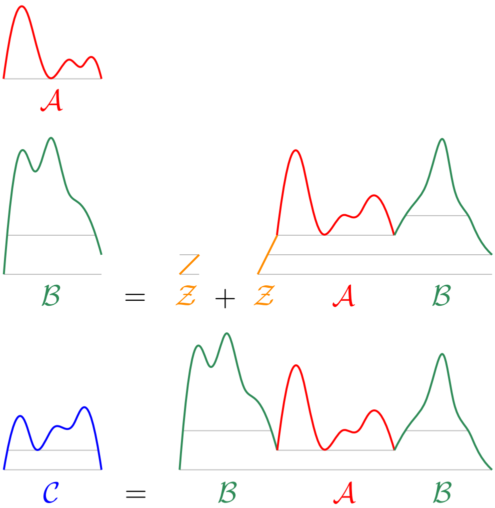
Elementary decomposition of basketball walks



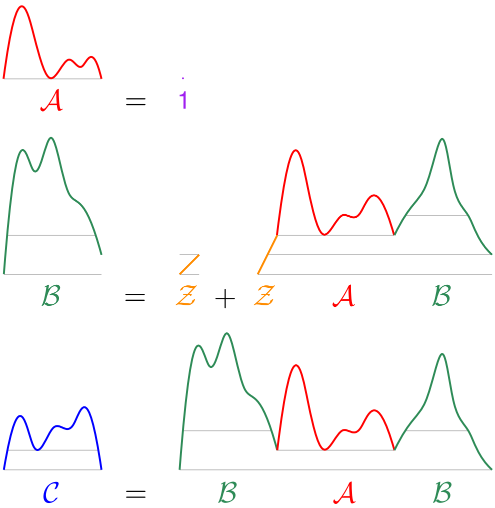
Elementary decomposition of basketball walks



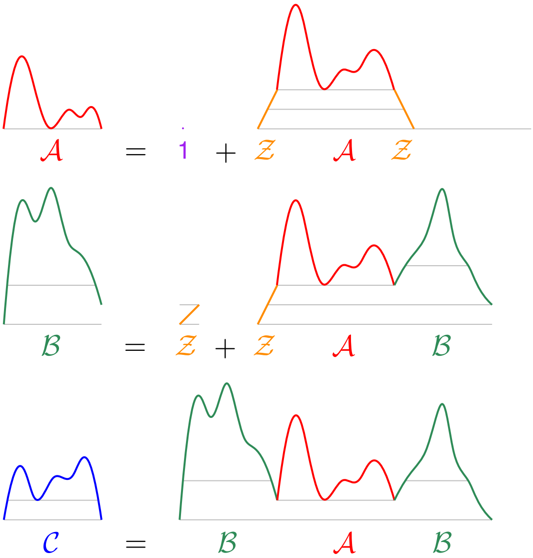
Elementary decomposition of basketball walks



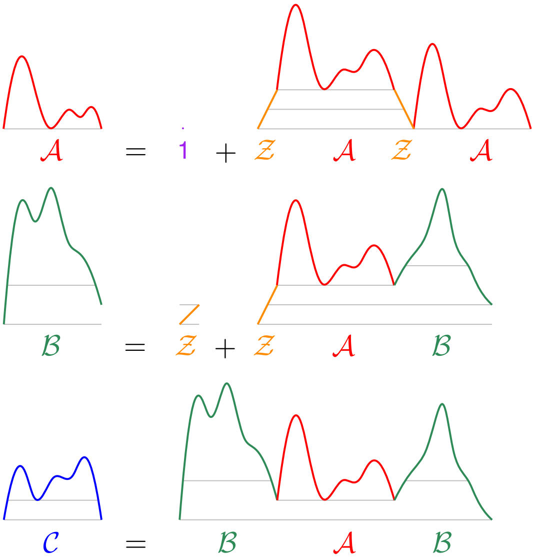
Elementary decomposition of basketball walks



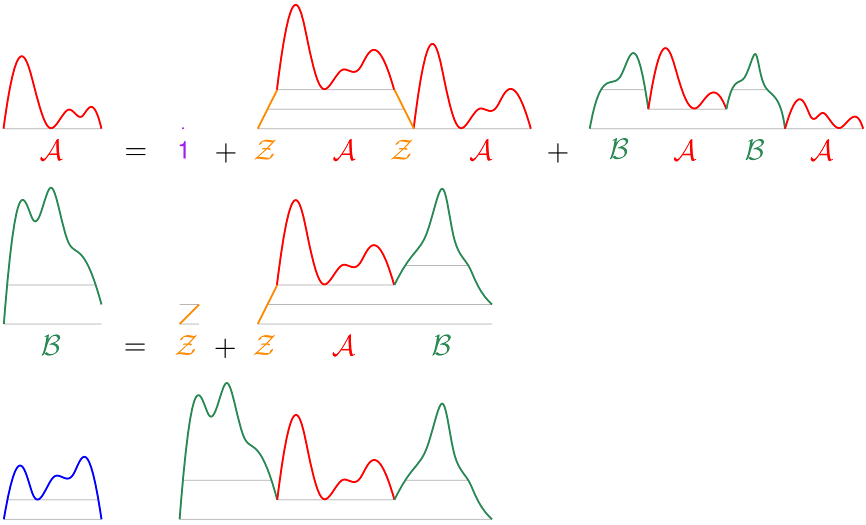
Elementary decomposition of basketball walks



Elementary decomposition of basketball walks



Elementary decomposition of basketball walks

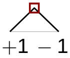


$$\mathcal{A} = 1 + \mathcal{Z}\,\mathcal{A}\,\mathcal{Z}\,\mathcal{A} + \mathcal{B}\,\mathcal{A}\,\mathcal{B}\,\mathcal{A}$$

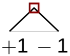
$$\mathcal{B} = \mathcal{Z} + \mathcal{Z}\,\mathcal{A}\,\mathcal{B}$$

$$\mathcal{C} = \mathcal{B}\,\mathcal{A}\,\mathcal{B}$$

The bijection

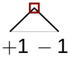


The bijection

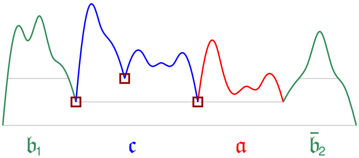


$$\overset{\Phi}{\longmapsto} \quad (\varepsilon, \varepsilon)$$

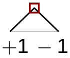
The bijection



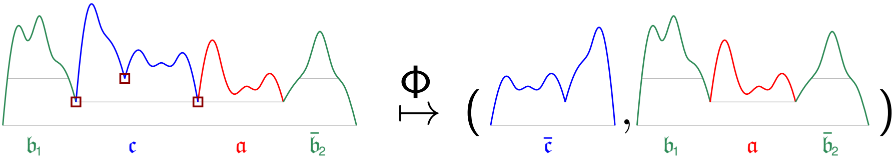
$$\overset{\Phi}{\longmapsto} \quad (\varepsilon, \varepsilon)$$



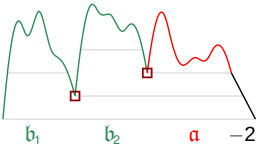
The bijection



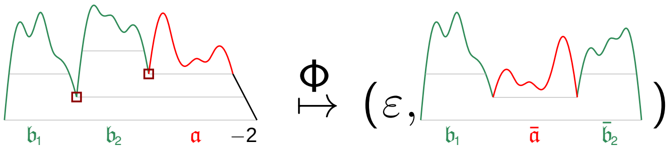
$$\overset{\Phi}{\mapsto} \quad (\varepsilon, \varepsilon)$$



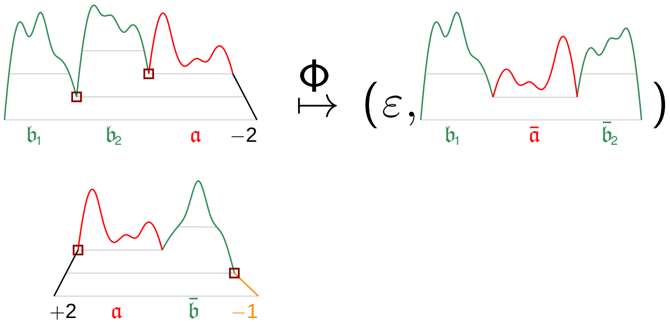
The bijection



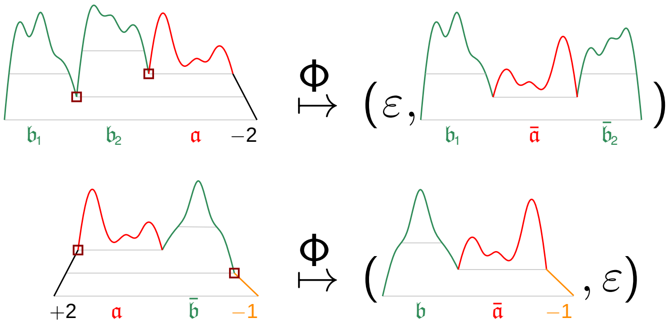
The bijection



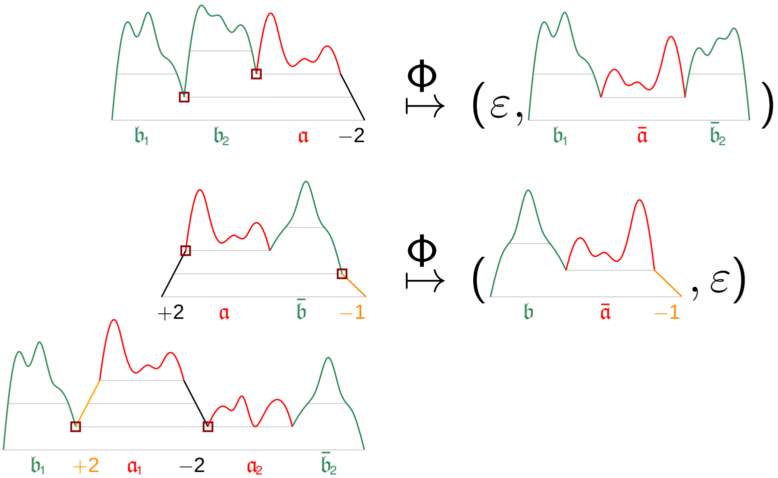
The bijection



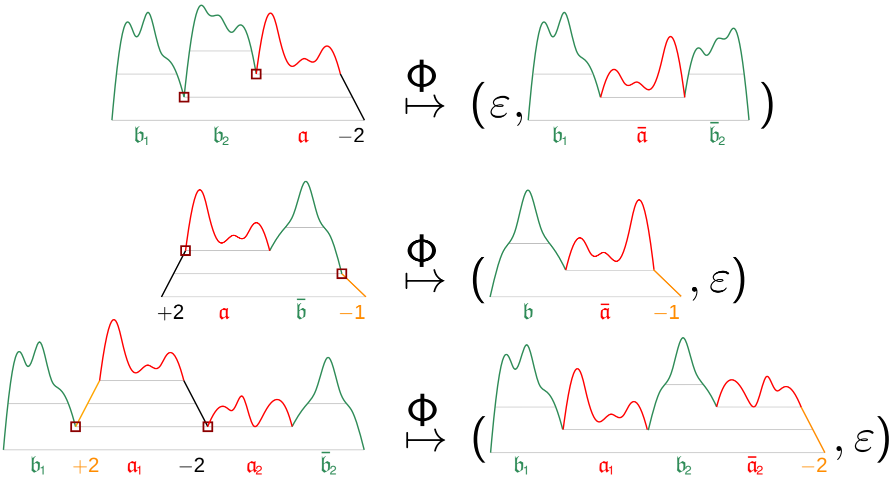
The bijection



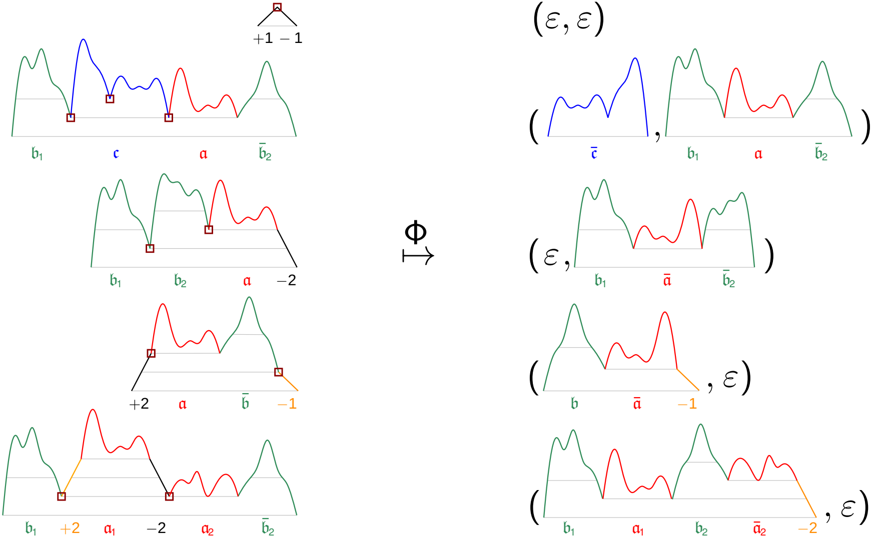
The bijection



The bijection

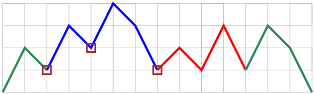


The bijection

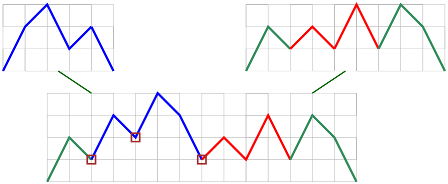


The bijection

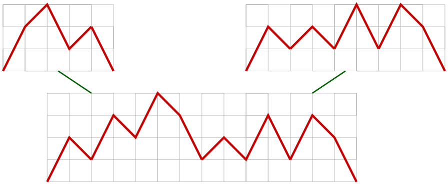
The bijection



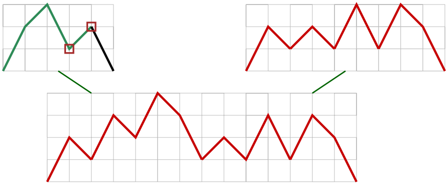
The bijection



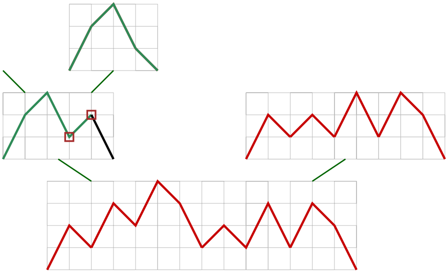
The bijection



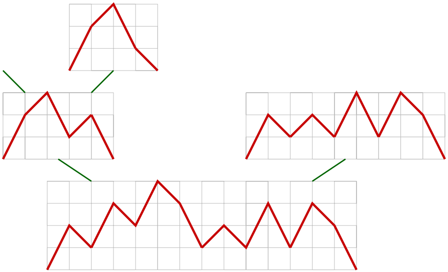
The bijection



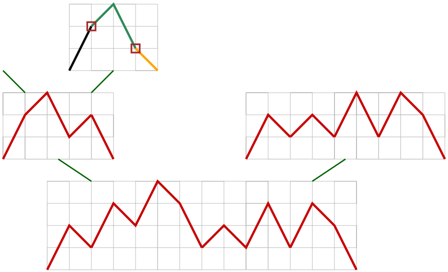
The bijection



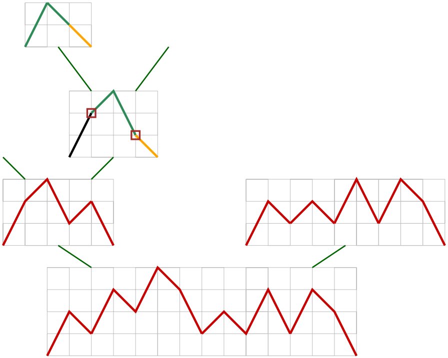
The bijection



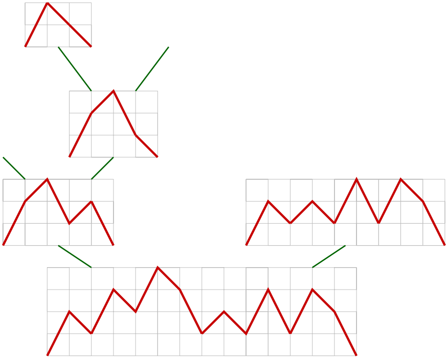
The bijection



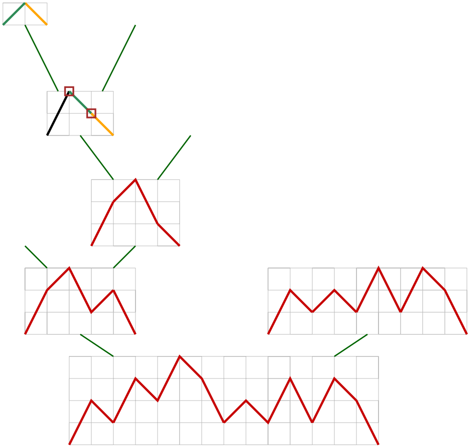
The bijection



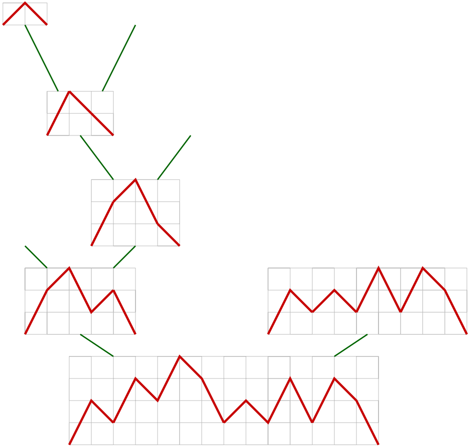
The bijection



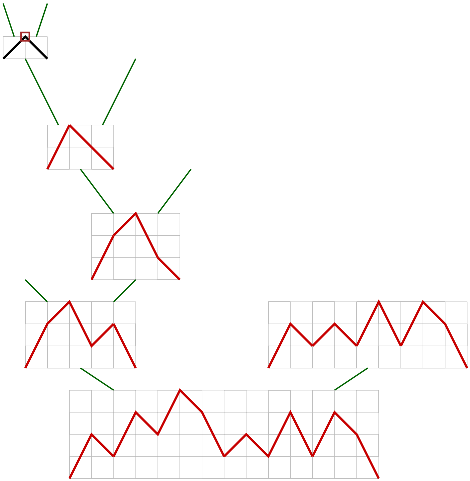
The bijection



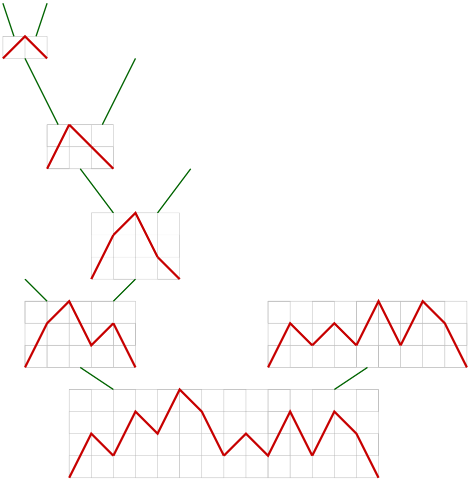
The bijection



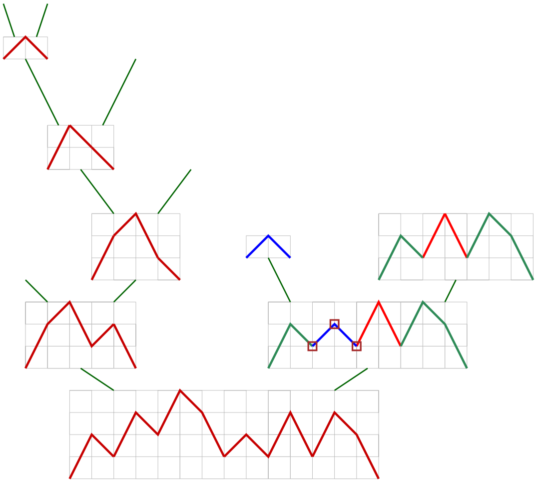
The bijection



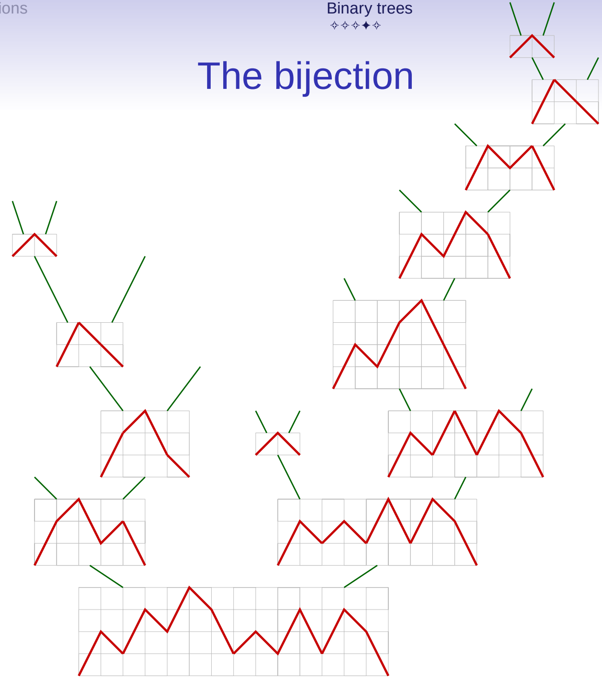
The bijection



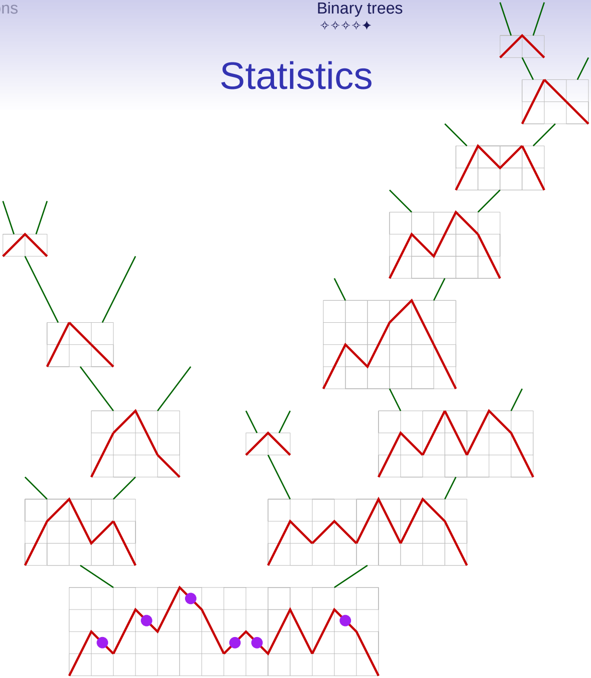
The bijection



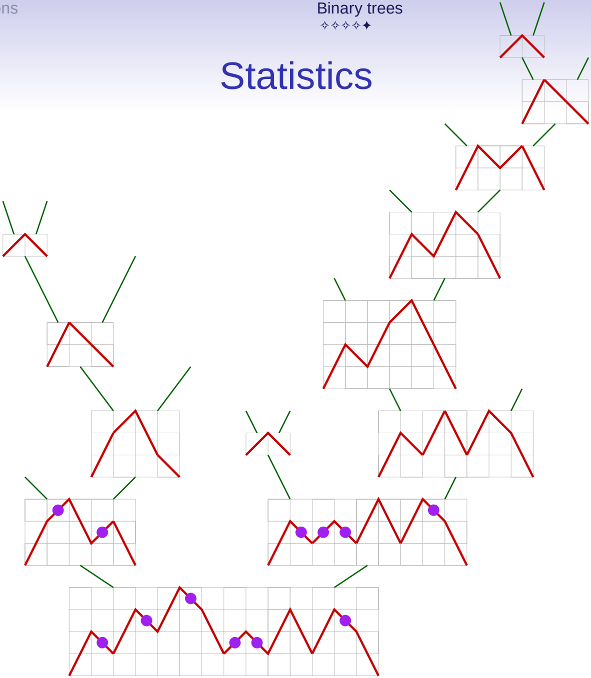
The bijection



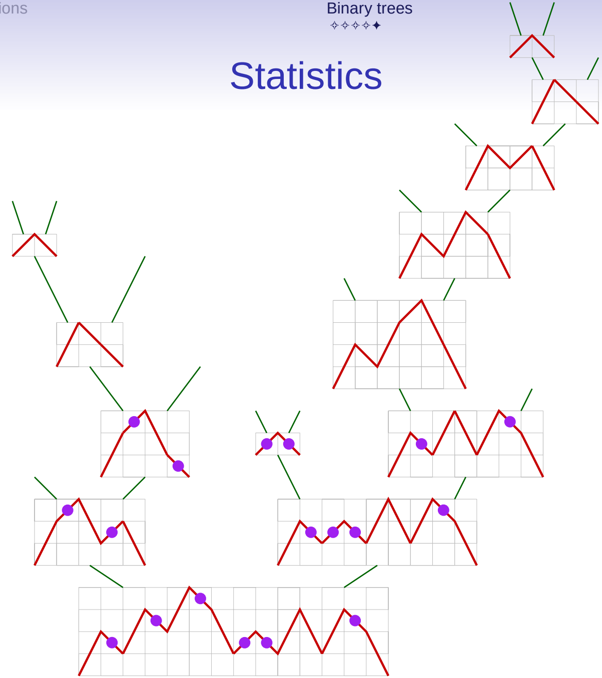
Statistics



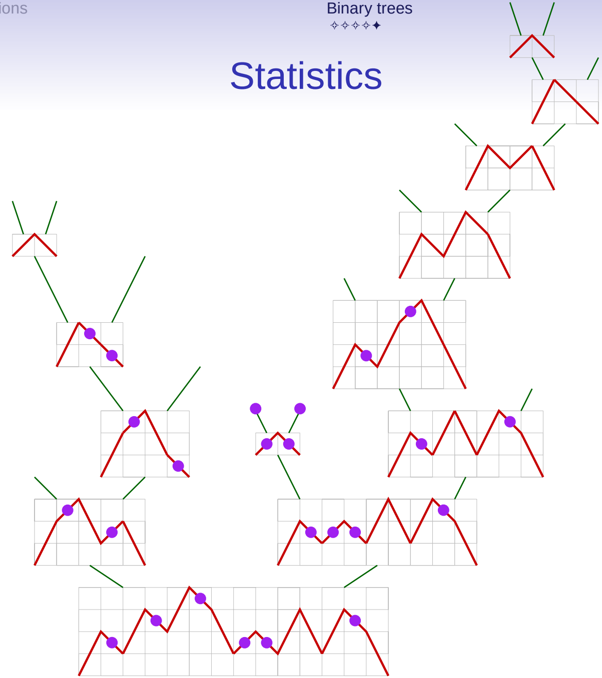
Statistics



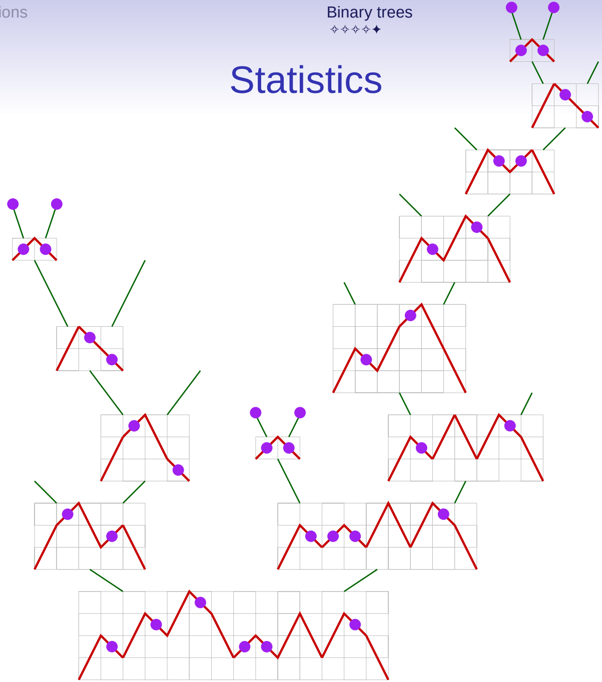
Statistics



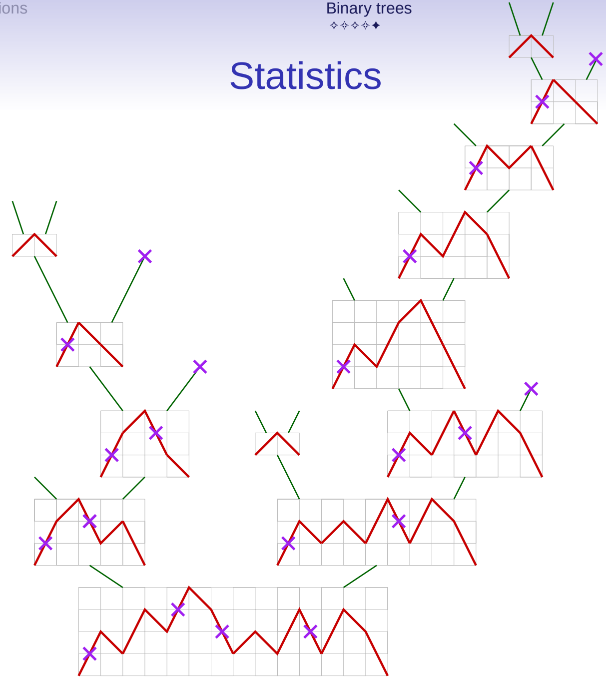
Statistics



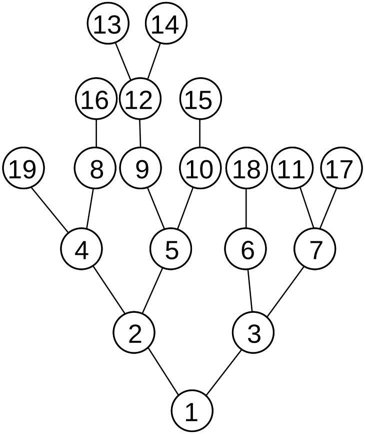
Statistics



Statistics



Valid permutations on a unary-binary tree

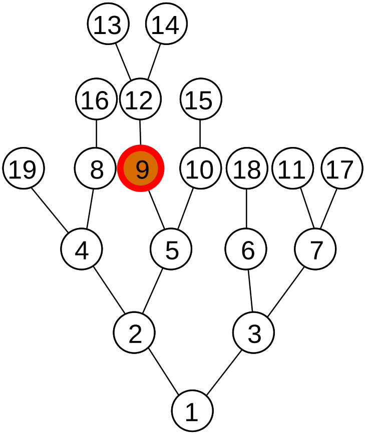


Lemma

A permutation is valid for a tree if and only if it avoids 213 and, the value taken at each node is a right-to-left minimum.

1 2 3 4 5 6 7 19 8 9 10 18 11 17 16 12 15 13 14

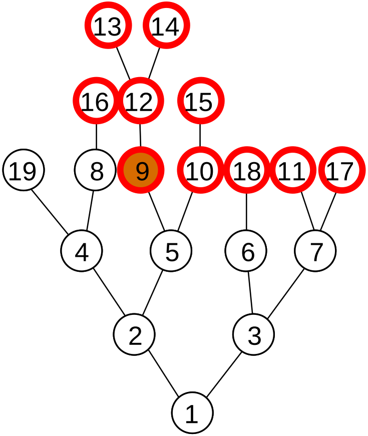
Valid permutations on a unary-binary tree



Lemma

A permutation is valid for a tree if and only if it avoids 213 and, the value taken at each node is a right-to-left minimum.

Valid permutations on a unary-binary tree

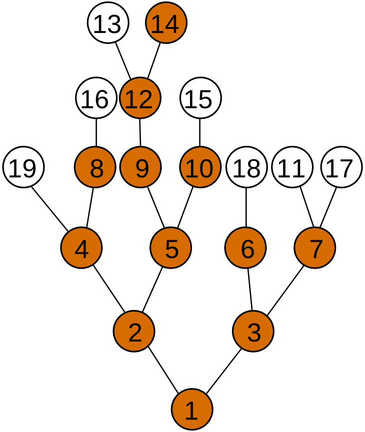


Lemma

A permutation is valid for a tree if and only if it avoids 213 and, the value taken at each node is a right-to-left minimum.

1 2 3 4 5 6 7 19 8 9 10 18 11 17 16 12 15 13 14

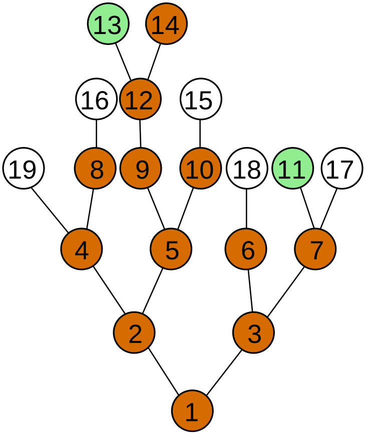
Valid permutations on a unary-binary tree



Lemma

A permutation is valid for a tree if and only if it avoids 213 and, the value taken at each node is a right-to-left minimum.

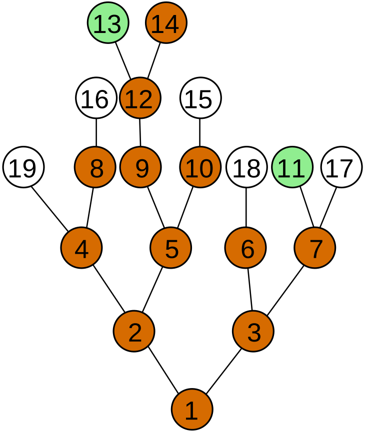
Valid permutations on a unary-binary tree

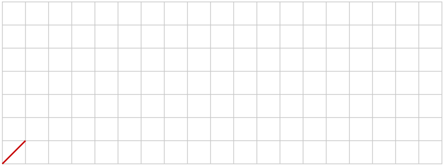


Lemma

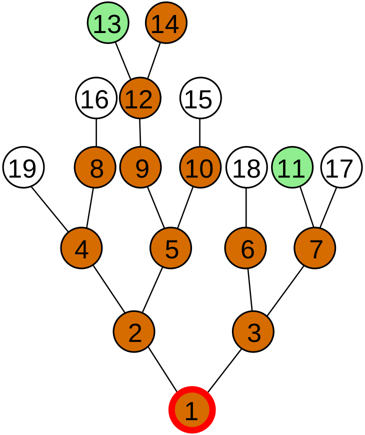
A permutation is valid for a tree if and only if it avoids 213 and, the value taken at each node is a right-to-left minimum.

Encoding by decorated Motzkin paths

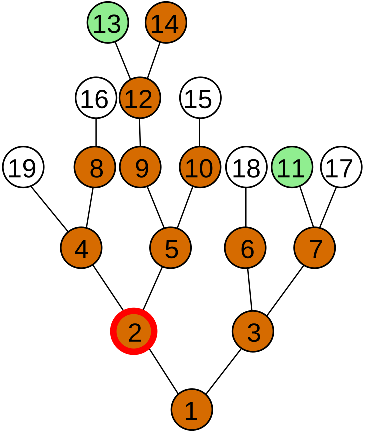


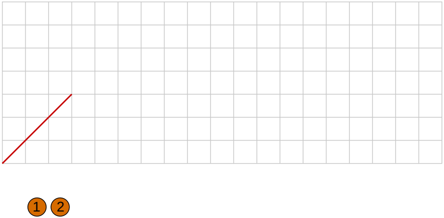


Encoding by decorated Motzkin paths

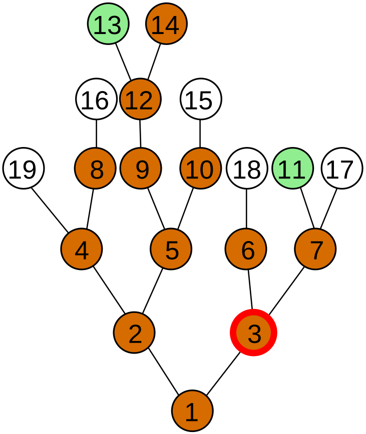


Encoding by decorated Motzkin paths

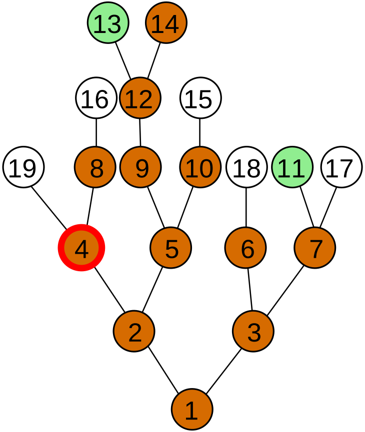


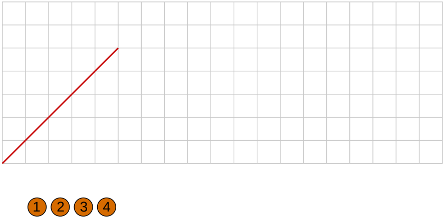


Encoding by decorated Motzkin paths

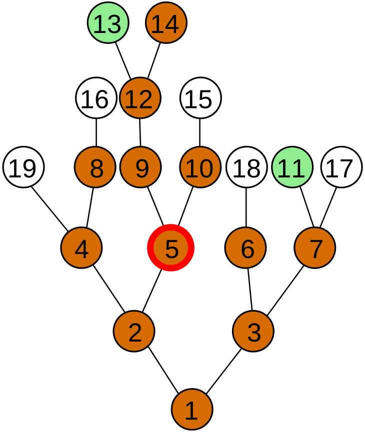


Encoding by decorated Motzkin paths

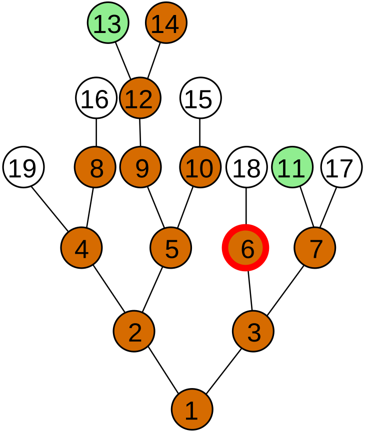


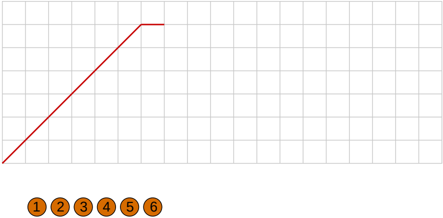


Encoding by decorated Motzkin paths

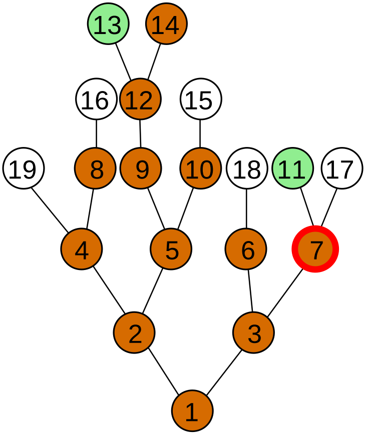


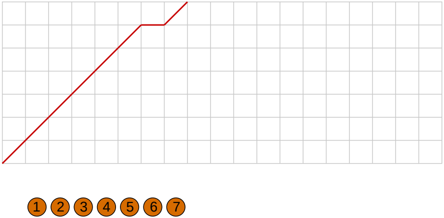
Encoding by decorated Motzkin paths



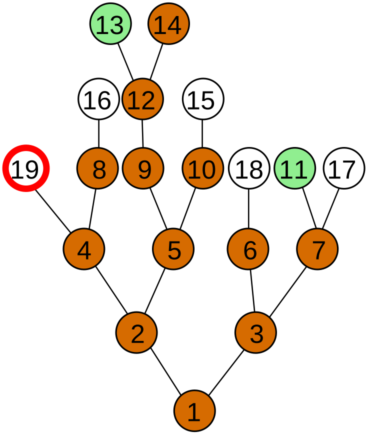


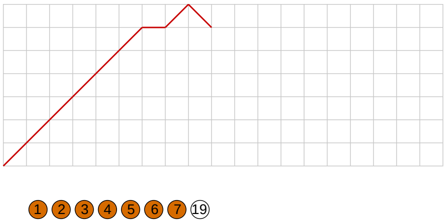
Encoding by decorated Motzkin paths



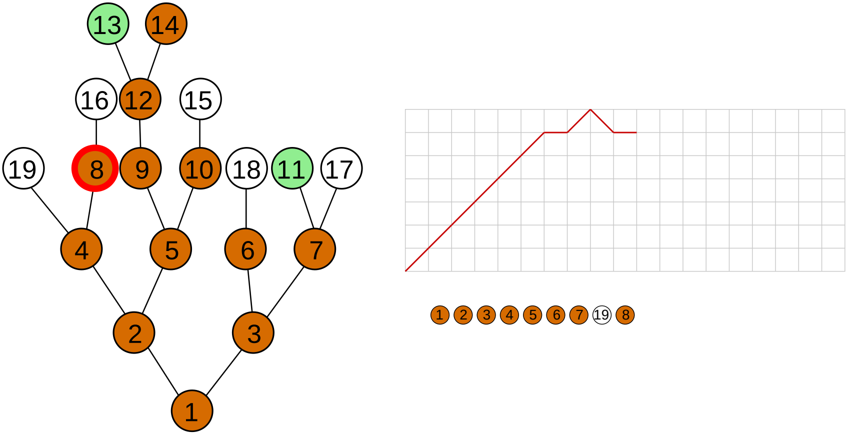


Encoding by decorated Motzkin paths

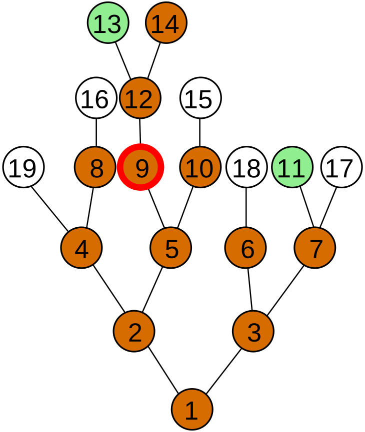


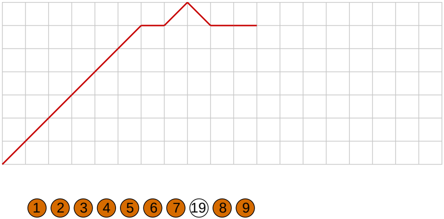


Encoding by decorated Motzkin paths

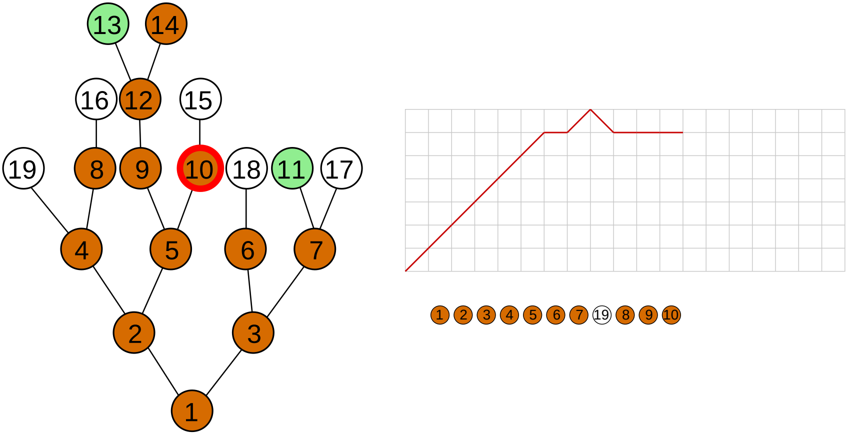


Encoding by decorated Motzkin paths

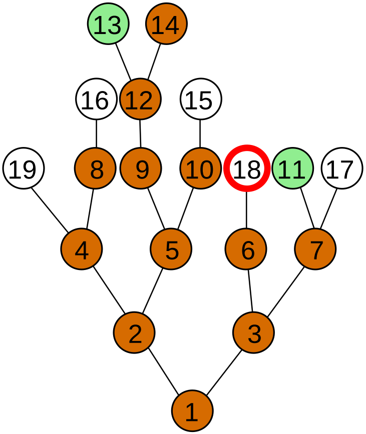


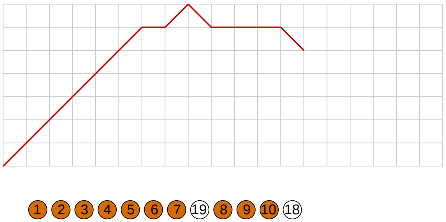


Encoding by decorated Motzkin paths

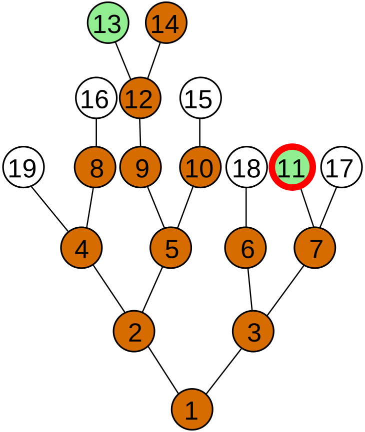


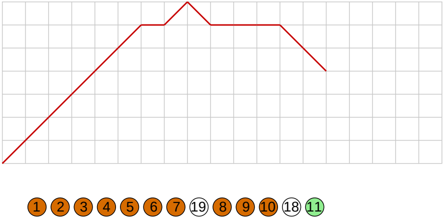
Encoding by decorated Motzkin paths



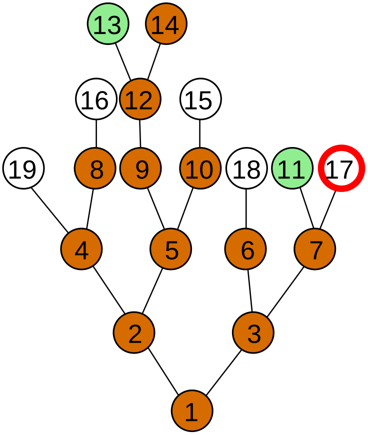


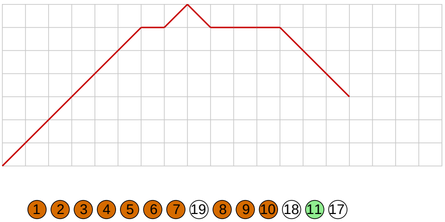
Encoding by decorated Motzkin paths



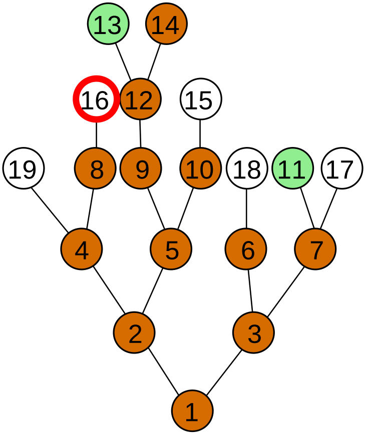


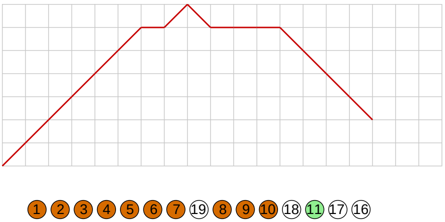
Encoding by decorated Motzkin paths



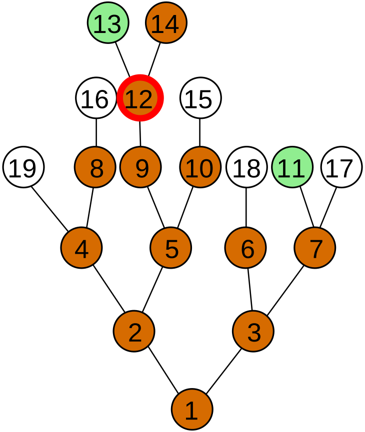


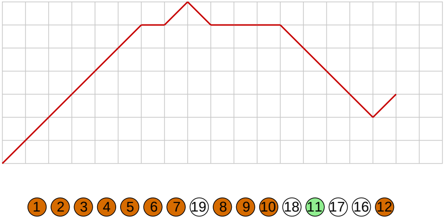
Encoding by decorated Motzkin paths



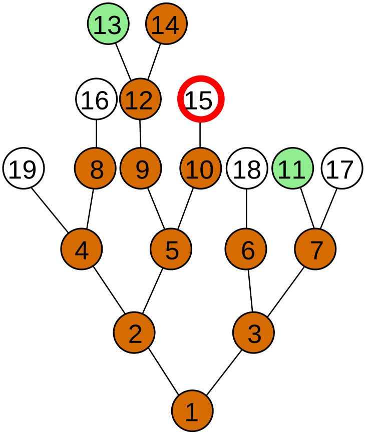


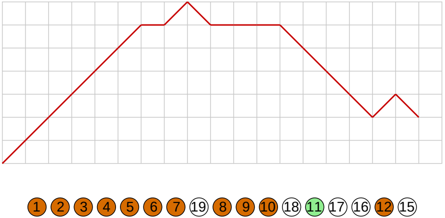
Encoding by decorated Motzkin paths



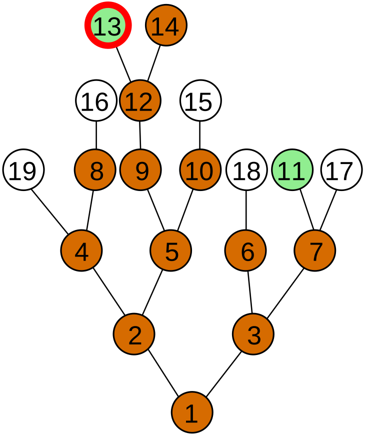


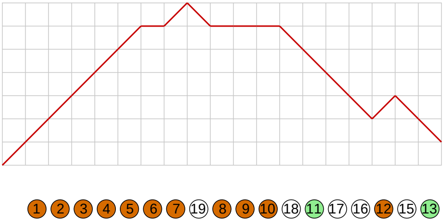
Encoding by decorated Motzkin paths



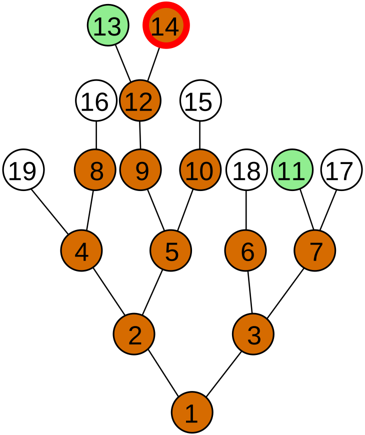


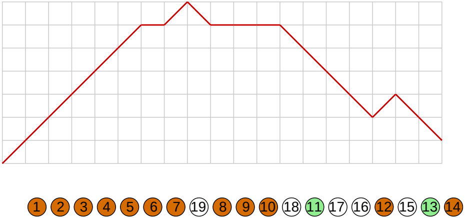
Encoding by decorated Motzkin paths



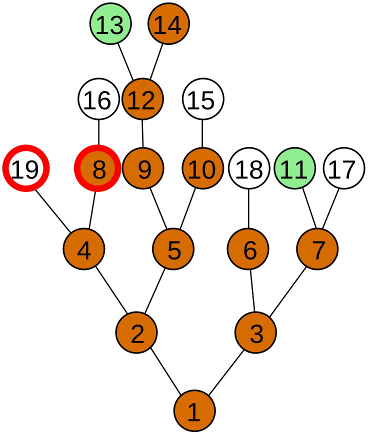


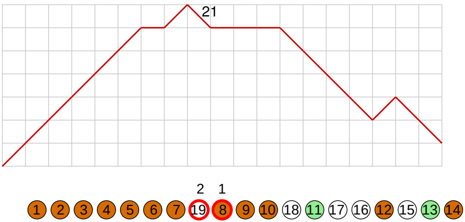
Encoding by decorated Motzkin paths



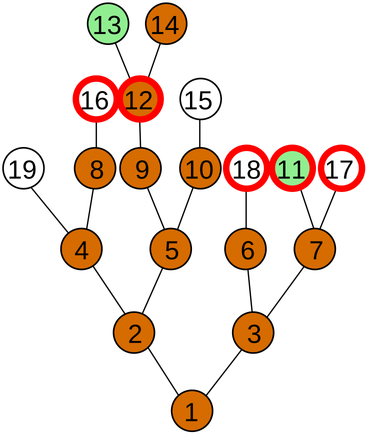


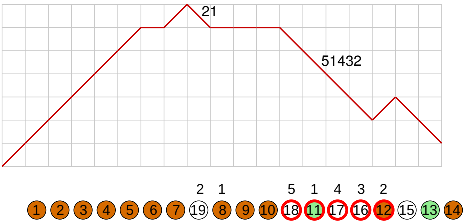
Encoding by decorated Motzkin paths



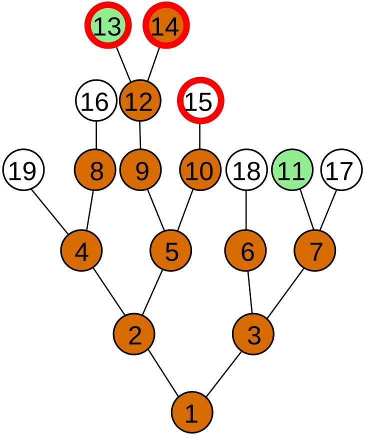


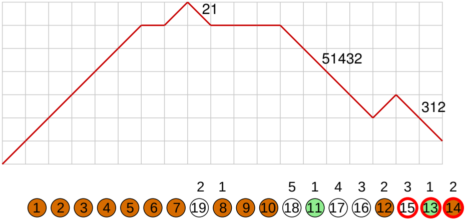
Encoding by decorated Motzkin paths



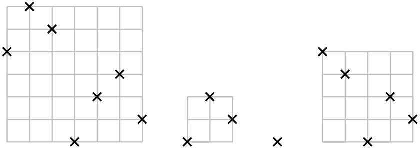


Encoding by decorated Motzkin paths

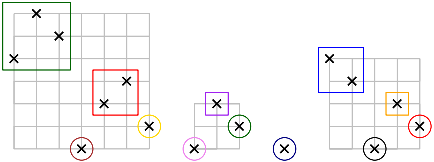




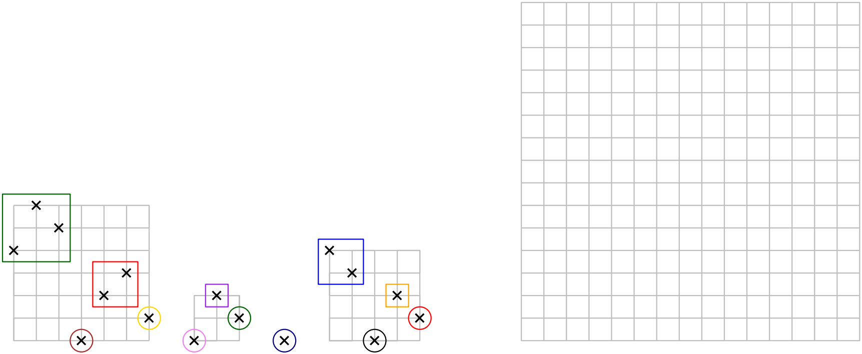
How to reconstruct the permutation



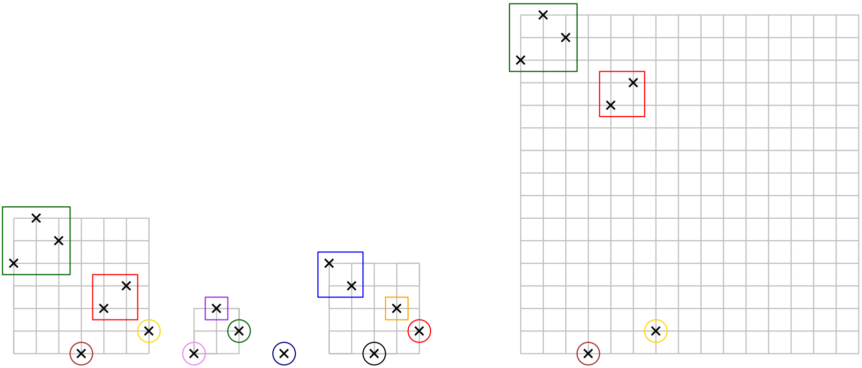
How to reconstruct the permutation



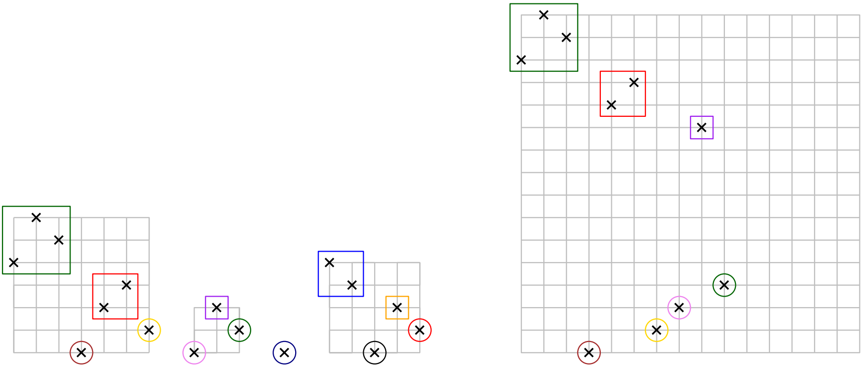
How to reconstruct the permutation



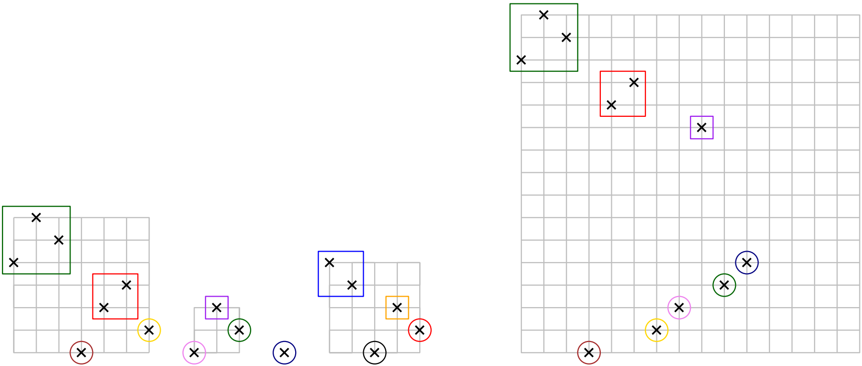
How to reconstruct the permutation



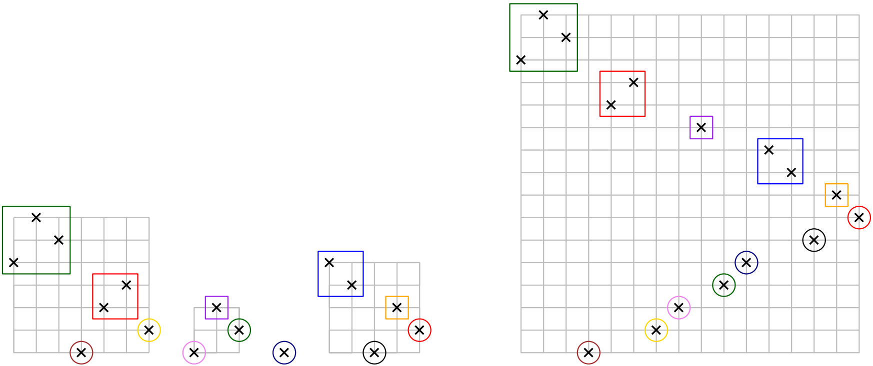
How to reconstruct the permutation



How to reconstruct the permutation



How to reconstruct the permutation

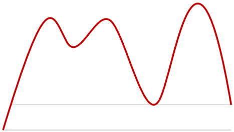


Counting decorated Motzkin walks

$\mathcal{T}$: decorated Motzkin walks

Aim

We want to show that $\mathcal{T} = \mathcal{G}$.



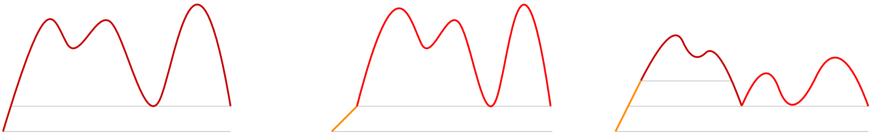
$\mathcal{G}$

Counting decorated Motzkin walks

$\mathcal{T}$: decorated Motzkin walks

Aim

We want to show that $\mathcal{T} = \mathcal{G}$.



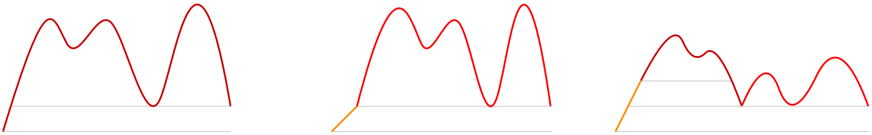
$$\mathcal{G} = \mathcal{Z}\ \mathcal{A} + \mathcal{Z}\ \mathcal{G}\ \mathcal{A}$$

Counting decorated Motzkin walks

$\mathcal{T}$: decorated Motzkin walks

Aim

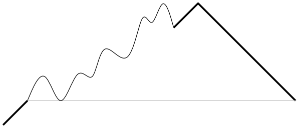
We want to show that $\mathcal{T} = \mathcal{G}$.



$$\mathcal{G} = \mathcal{Z} \quad \mathcal{A} + \mathcal{Z} \quad \mathcal{G} \quad \mathcal{A}$$

$$\mathcal{G} = (\mathcal{Z}\mathcal{A}) + (\mathcal{Z}\mathcal{A})^2 + (\mathcal{Z}\mathcal{A})^3 + \ldots = \mathrm{Seq}_{\geq 1}(\mathcal{Z}\mathcal{A})$$

Step 1: $\mathcal{T} = \mathrm{Seq}_{\geq 1}(\mathcal{M})$

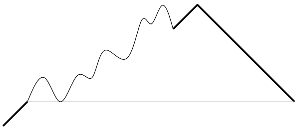


$\mathcal{M}$: decorated Motzkin walks whose last 0-step or $+1$-step is a $+1$-step

Claim

$$\mathcal{T} = \mathrm{Seq}_{\geq 1}(\mathcal{M})$$

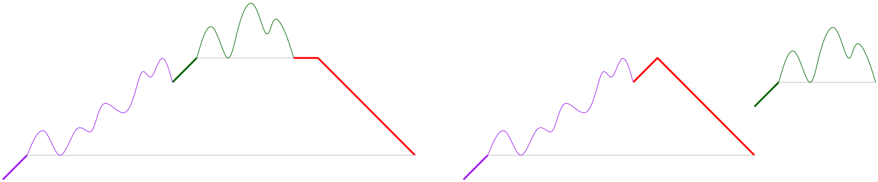
Step 1: $\mathcal{T} = \mathrm{Seq}_{\geq 1}(\mathcal{M})$



$\mathcal{M}$: decorated Motzkin walks whose last 0-step or $+1$-step is a $+1$-step

Claim

$$\mathcal{T} = \mathrm{Seq}_{\geq 1}(\mathcal{M})$$



$$\mathcal{T} - \mathcal{M} \quad = \quad \mathcal{M} \quad \times \quad \mathcal{T}$$

Step 2: $\mathcal{M} = \mathcal{Z}\mathcal{A}$

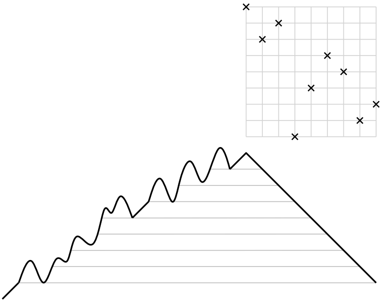
We observed that $B = z/(1 - zA)$ and $A = 1 + (zA)^2 + (BA)^2$. Thus,

$$zA = z\left(1 + (zA)^2 + \frac{(zA)^2}{(1 - zA)^2}\right)$$

Step 2: $\mathcal{M} = \mathcal{Z}\mathcal{A}$

We observed that $B = z/(1 - zA)$ and $A = 1 + (zA)^2 + (BA)^2$. Thus,

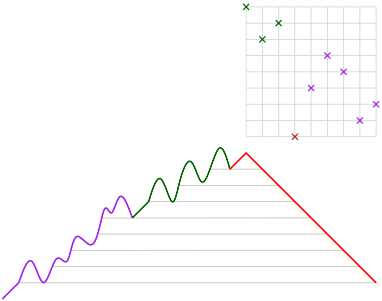
$$zA = z\left(1 + (zA)^2 + \frac{(zA)^2}{(1 - zA)^2}\right)$$



Step 2: $\mathcal{M} = \mathcal{Z}\mathcal{A}$

We observed that $B = z/(1 - zA)$ and $A = 1 + (zA)^2 + (BA)^2$. Thus,

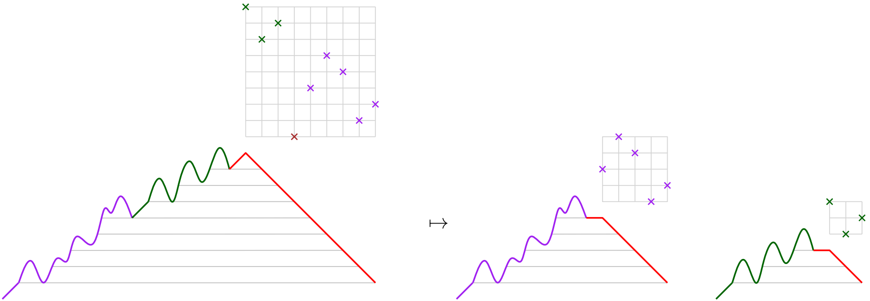
$$zA = z\left(1 + (zA)^2 + \frac{(zA)^2}{(1-zA)^2}\right)$$



Step 2: $\mathcal{M} = \mathcal{Z}\mathcal{A}$

We observed that $B = z/(1 - zA)$ and $A = 1 + (zA)^2 + (BA)^2$. Thus,

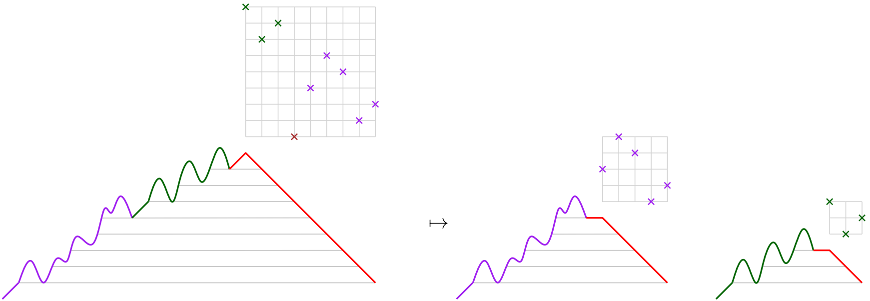
$$zA = z\left(1 + (zA)^2 + \frac{(zA)^2}{(1 - zA)^2}\right)$$



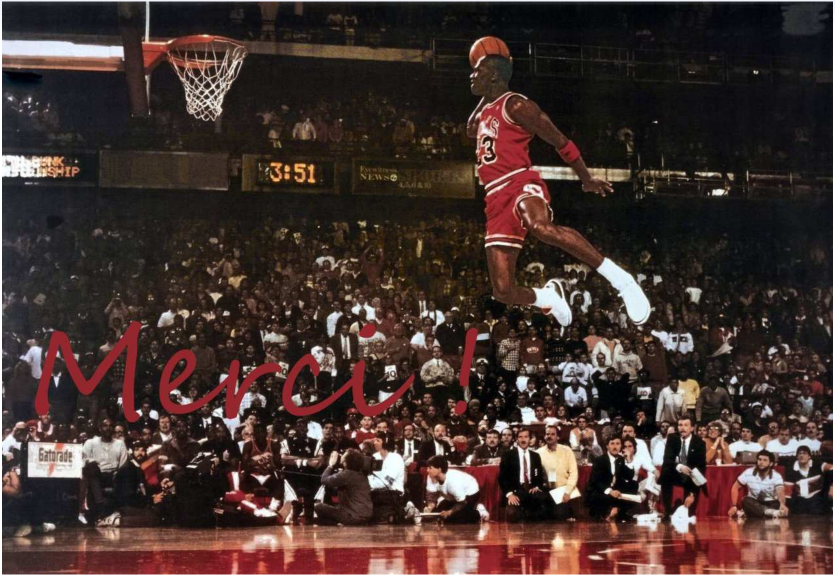
Step 2: $\mathcal{M} = \mathcal{Z}\mathcal{A}$

We observed that $B = z/(1 - zA)$ and $A = 1 + (zA)^2 + (BA)^2$. Thus,

$$zA = z\left(1 + (zA)^2 + \frac{(zA)^2}{(1 - zA)^2}\right)$$



$$M = z\,(1 + T - M)^2 = z\left(1 - M + \frac{M}{1 - M}\right)^2 = z\left(1 + M^2 + \frac{M^2}{(1 - M)^2}\right)$$


Merci !
3:51
Gatorade