COMBS

Sophia Yakoubov
(sonka89@mit.edu)

Motivation

Cool problem: enumerating pattern avoiding linear extensions of posets

Motivation

Cool problem: enumerating pattern avoiding linear extensions of posets

Hard for arbitrary posets!

Motivation

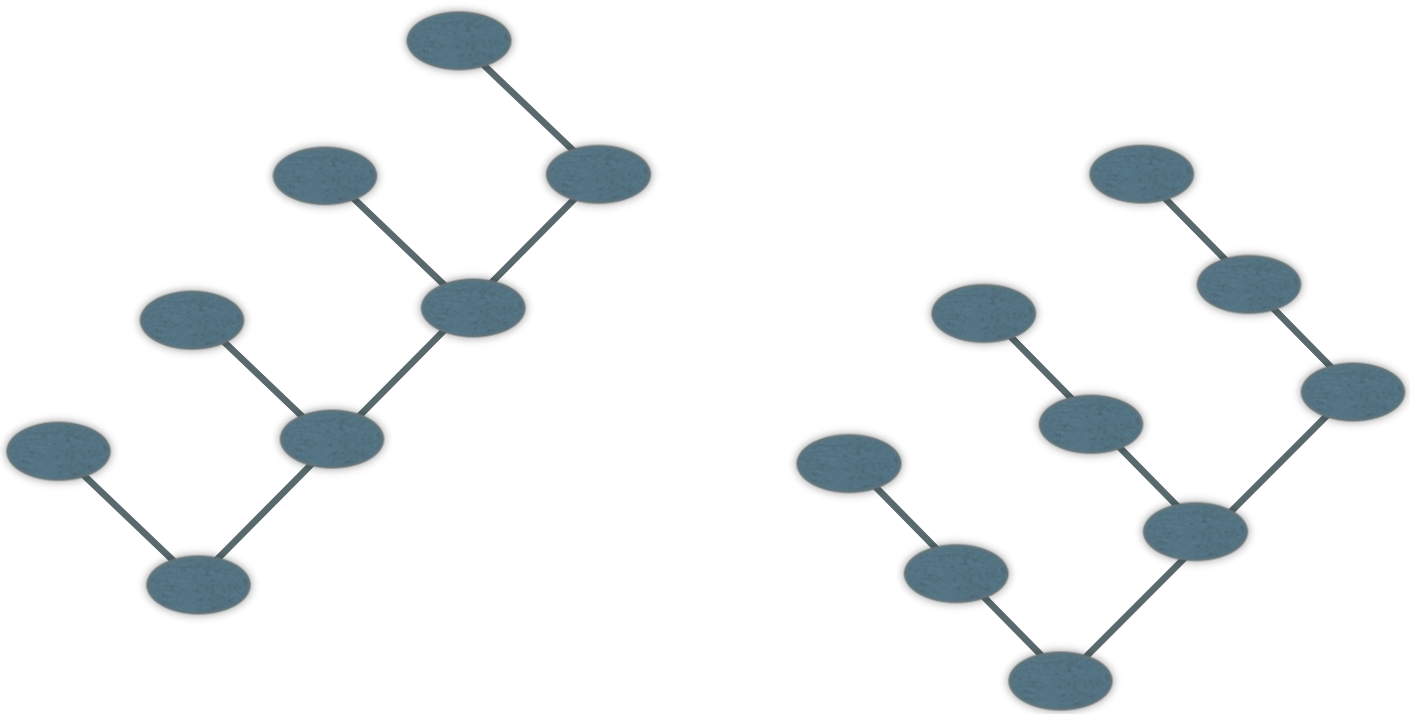
Cool problem: enumerating pattern avoiding linear extensions of posets

Hard for arbitrary posets!

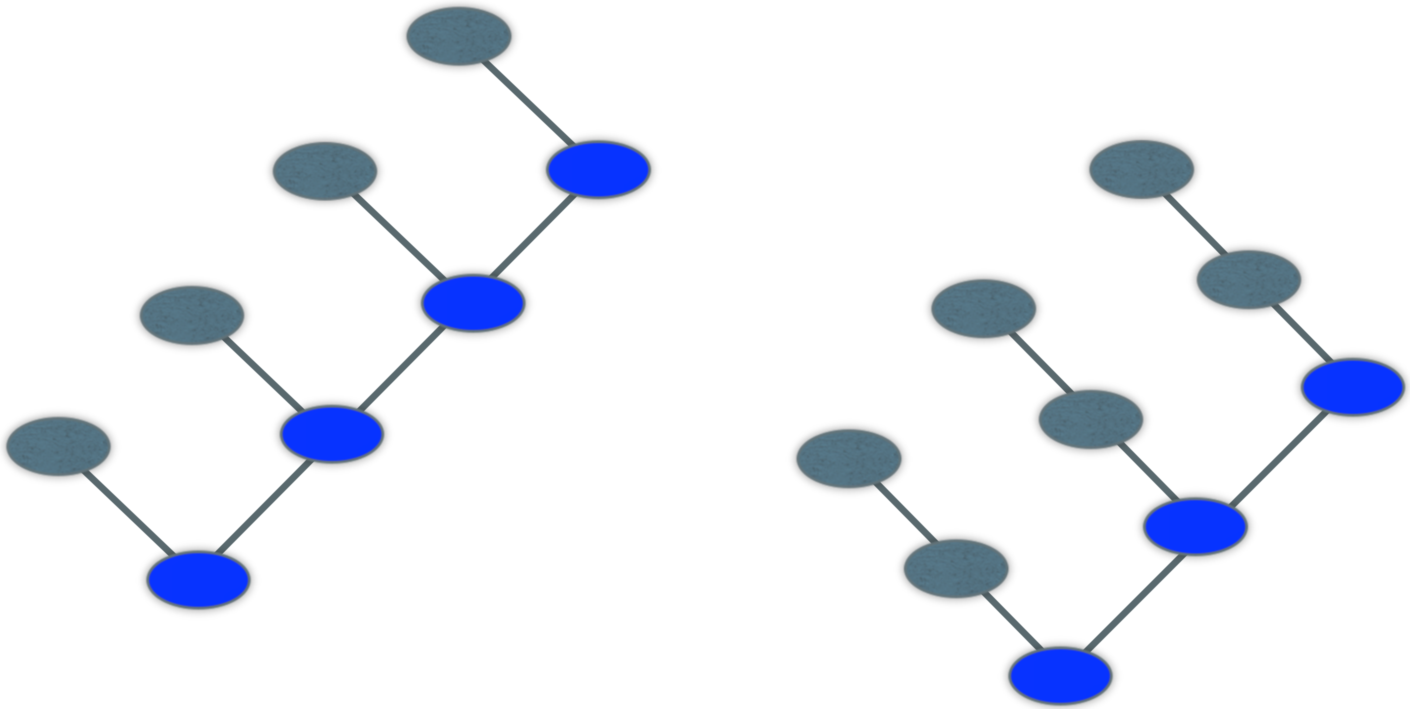
Narrowing the scope to specific category of posets makes it more feasible

Combs

Combs (the Posets, not the Styling Devices!)

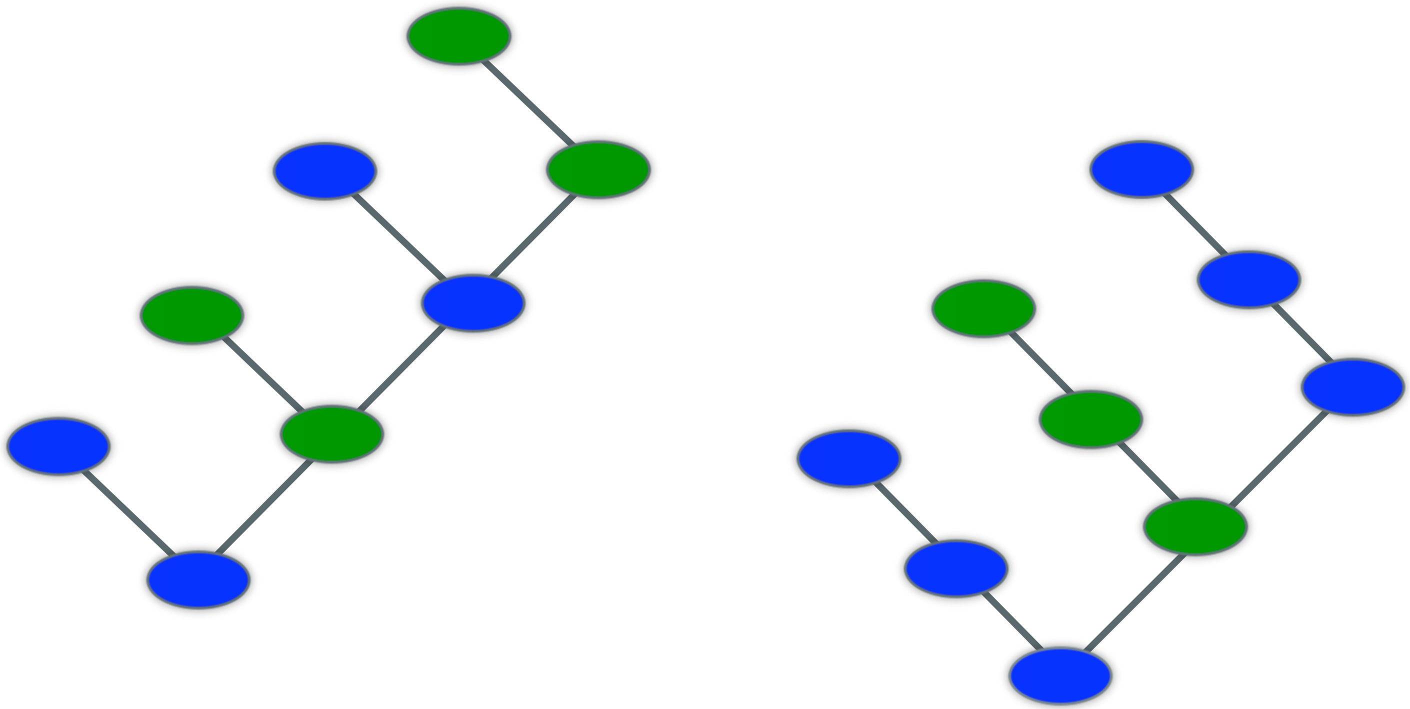


Combs (the Posets, not the Styling Devices!)



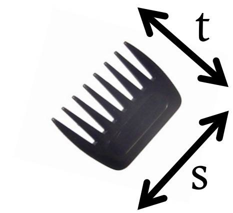
Comb Spines

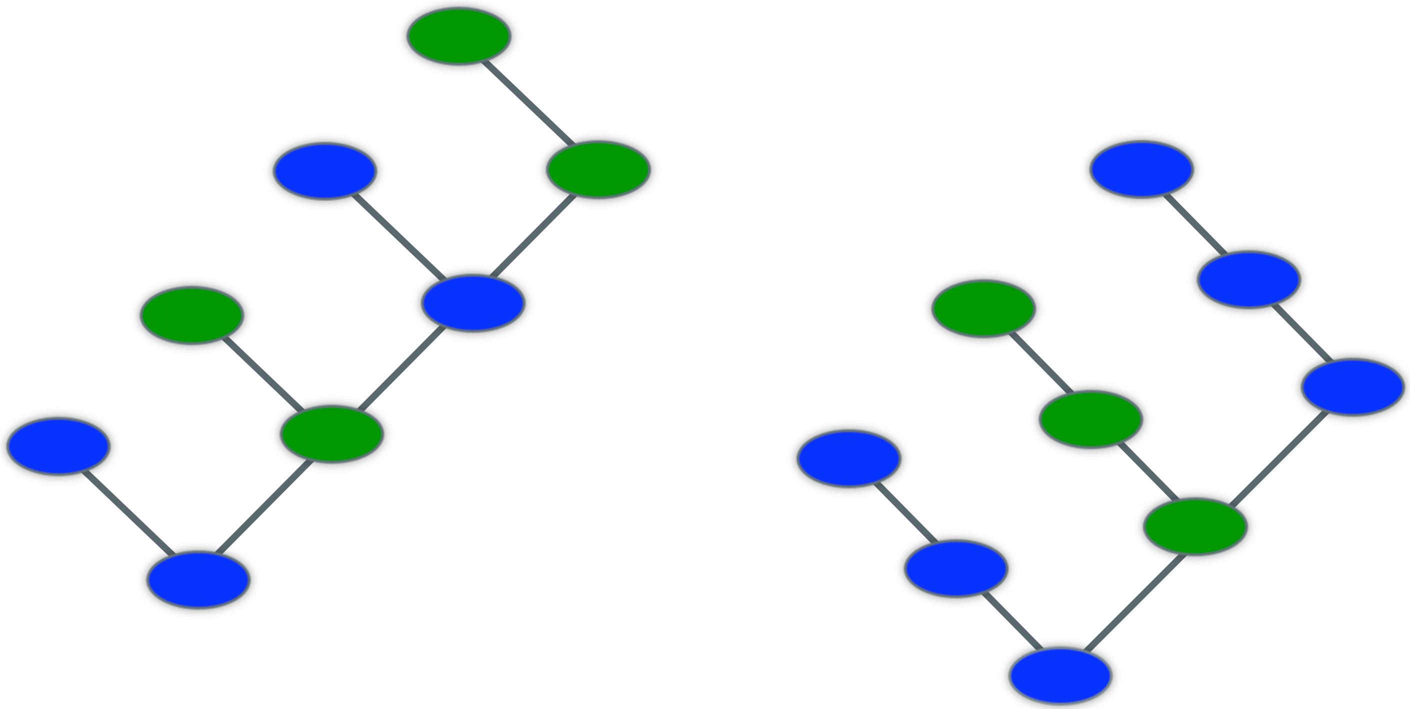
Combs (the Posets, not the Styling Devices!)



Comb Teeth

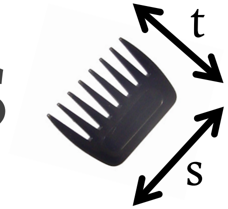
Combs (the Posets, not the Styling Devices!)

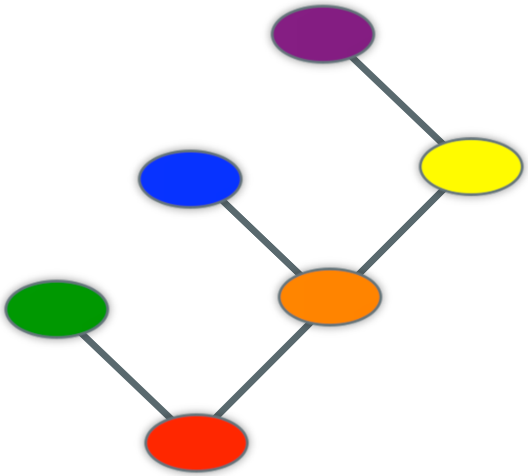




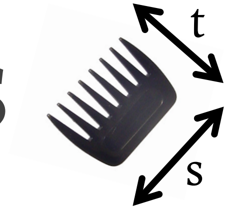
Comb Teeth

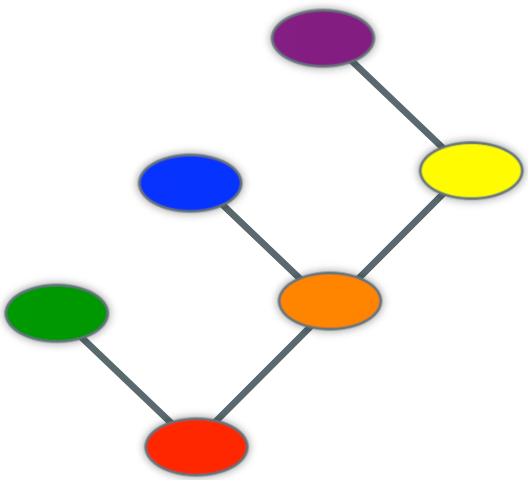
Comb Linear Extensions

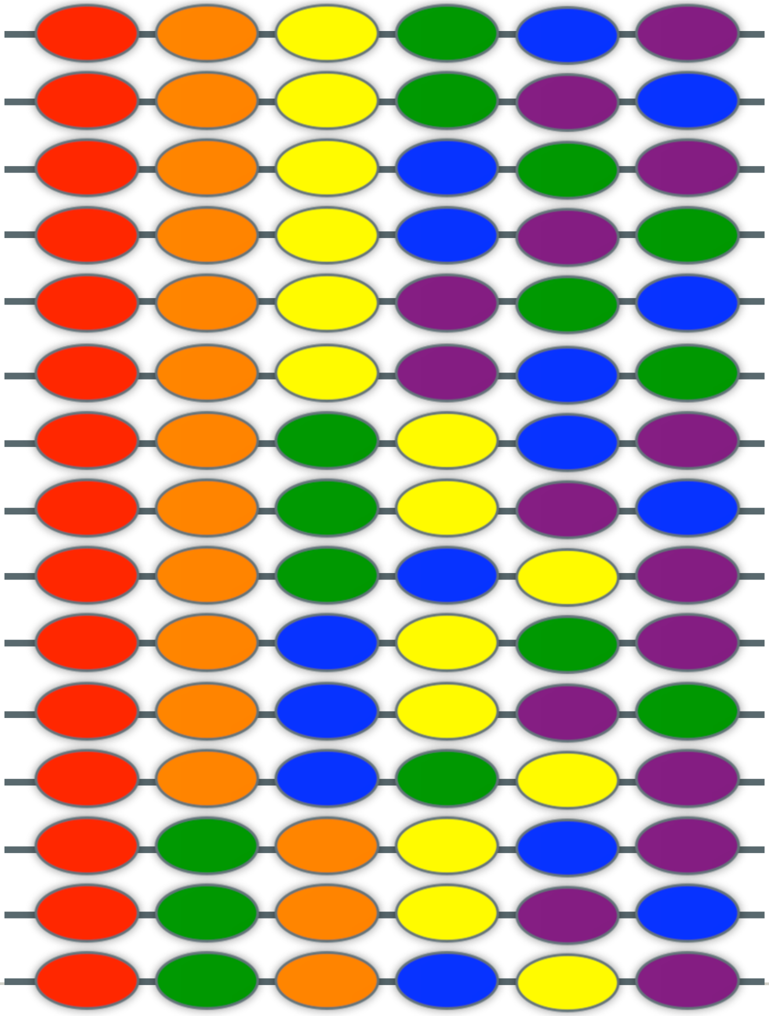




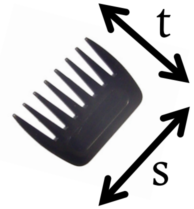
Comb Linear Extensions







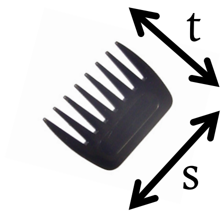
Enumerating Comb Linear Extensions



Theorem (Knuth): Let P be an n-element poset whose Hasse diagram forms a rooted tree such that the ith element has d_i descendants including itself. P then has E(P) extensions, where E(P) is defined as follows:

$$E(P) = \frac{n!}{\prod_{i \in [n]} d_i}$$

Enumerating Comb Linear Extensions

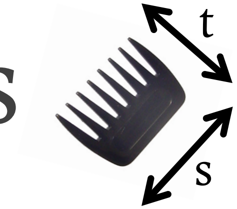


Theorem (Knuth): Let *P* be an *n*-element poset whose Hasse diagram forms a rooted tree such that the *i*th element has d_i descendants including itself. *P* then has E(*P*) extensions, where E(*P*) is defined as follows:

$$E(P) = \frac{n!}{\prod_{i \in [n]} d_i}$$

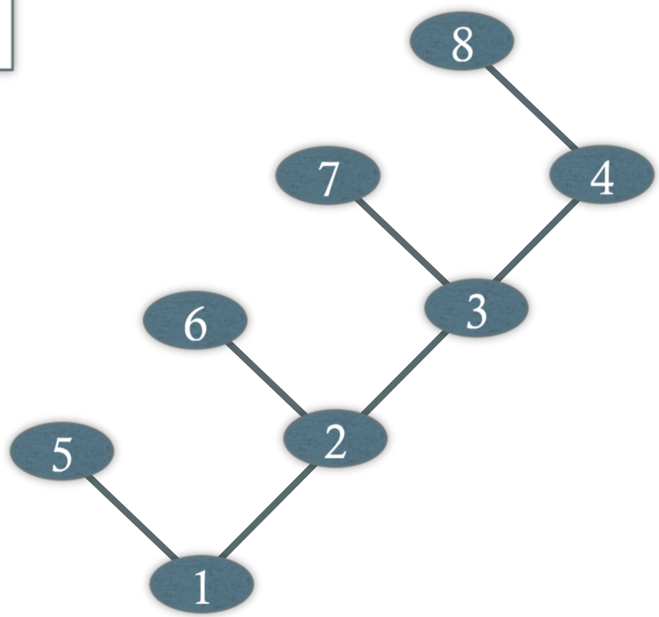
It follows that a comb has $\dfrac{(st)!}{s!(t!)^s}$ extensions.

Type- α and Type-β Combs

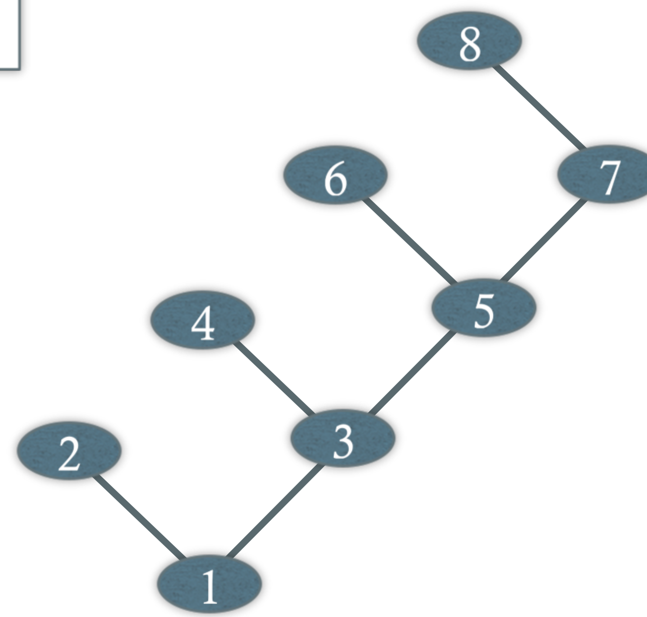


We assign integers to comb elements in two natural ways:

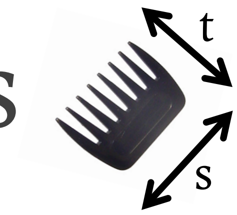
α



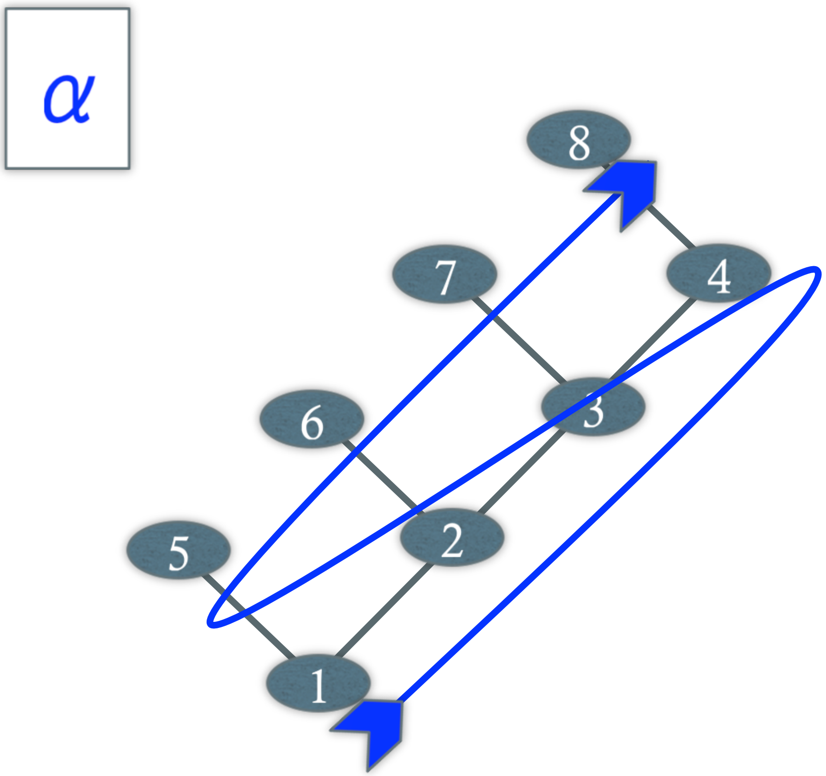
β

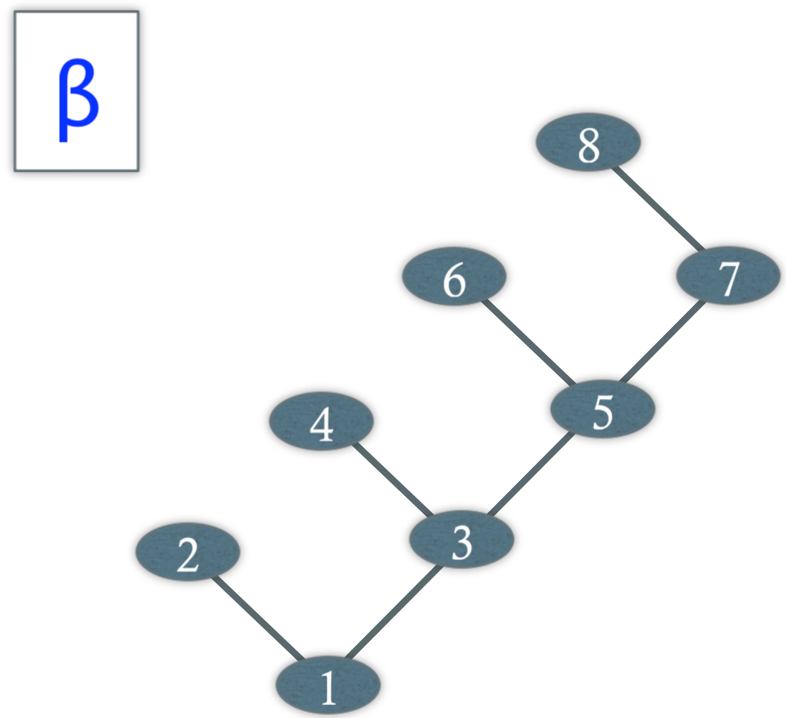


Type-α and Type-β Combs

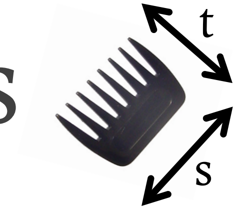


We assign integers to comb elements in two natural ways:

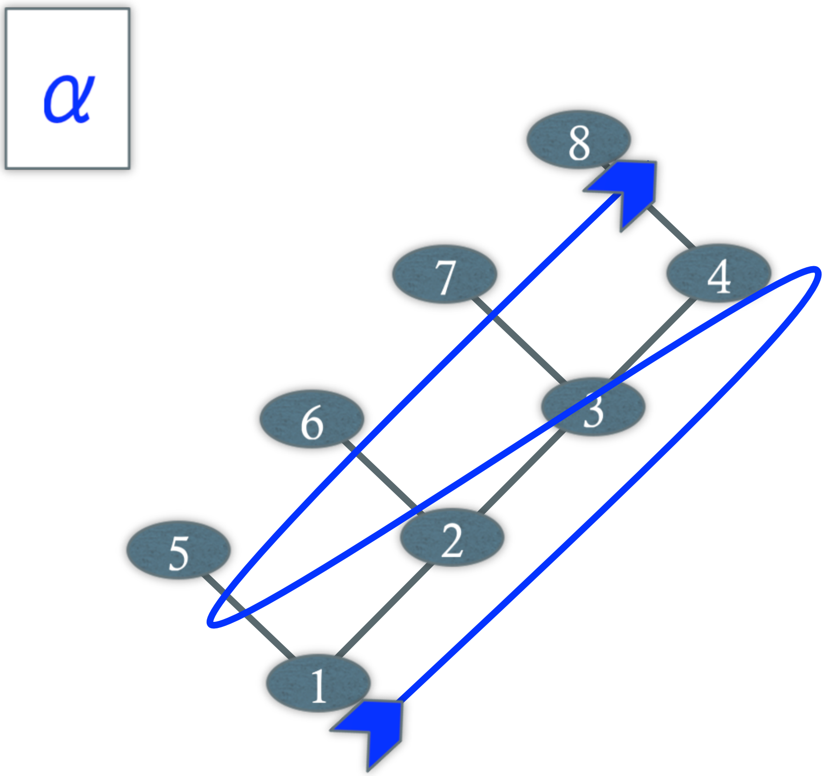


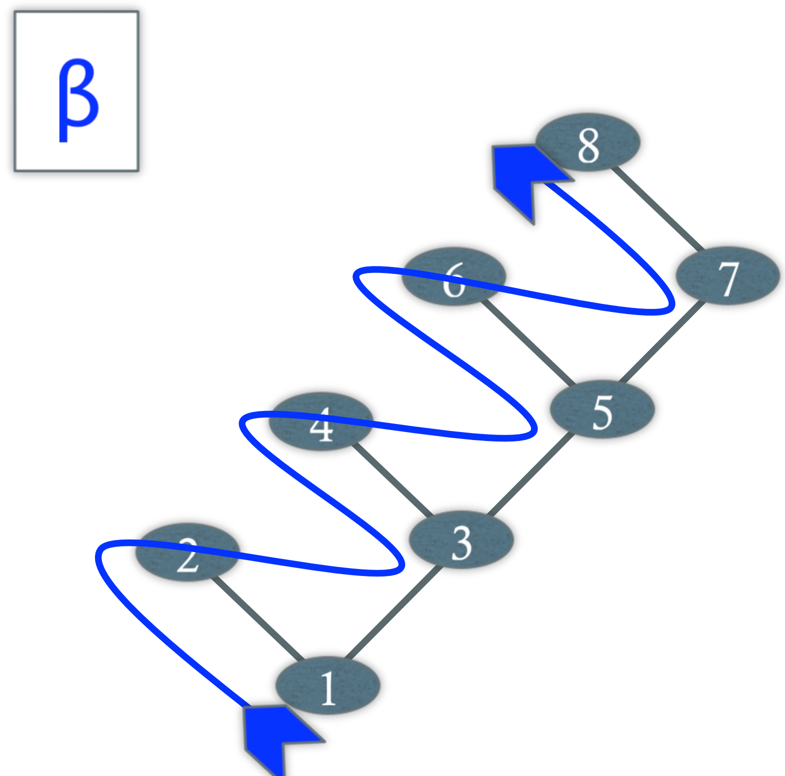


Type- α and Type-β Combs

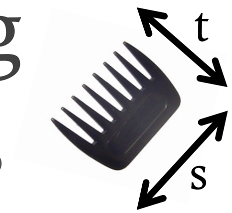


We assign integers to comb elements in two natural ways:



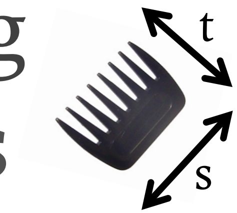


Enumeration of Pattern Avoiding Type-α and β Comb Extensions



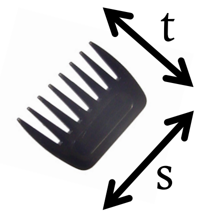
Pattern	Number of Avoiding Type-α Comb Extensions	Number of Avoiding Type-β Comb Extensions
123	0	0
132	1	1
213	Catalan Numbers (C_s)	t^{s-1}
231	Open	t^{s-1}
312	C_{s+1} - C_s	$\frac{1}{ts+1}\binom{s(t+1)}{s}$
321	Open	Open for $t > 2$

Enumeration of Pattern Avoiding Type-α and β Comb Extensions



Pattern	Number of Avoiding Type-α Comb Extensions	Number of Avoiding Type-β Comb Extensions
123	0	0
132	1	1
213	Catalan Numbers (C_s)	t^{s-1}
231	Open	t^{s-1}
312	C_{s+1} - C_s	$\frac{1}{ts+1}\binom{s(t+1)}{s}$
321	Open	Open for t > 2

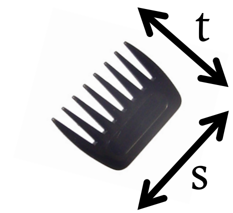
Type-β 312-Avoidance



Number of Type-β Comb Extensions Avoiding 312			
s \ t	2	3	4
1	1	1	1
2	3	4	5
3	12	22	35
4	55	140	285
5	273	969	2530
6	1428	7084	23751

Theorem: The number of Type-β comb extensions avoiding 312 is $\frac{1}{ts+1}\binom{s(t+1)}{s}$.

Type-β 312-Avoidance



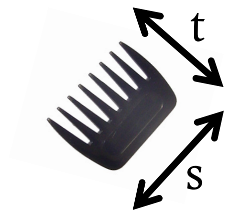
Number of Type-β Comb Extensions Avoiding 312			
s \ t	2	3	4
1	1	1	1
2	3	4	5
3	12	22	35
4	55	140	285
5	273	969	2530
6	1428	7084	23751

Theorem: The number of Type-β comb extensions avoiding 312 is $\frac{1}{ts+1}\binom{s(t+1)}{s}$.

This number is also:

- The number of $(t+1)$-ary trees on s nodes

Type-β 312-Avoidance



Number of Type-β Comb Extensions Avoiding 312

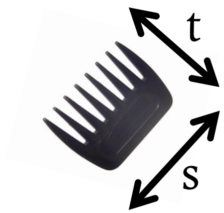
s \ t	2	3	4
1	1	1	1
2	3	4	5
3	12	22	35
4	55	140	285
5	273	969	2530
6	1428	7084	23751

Theorem: The number of Type-β comb extensions avoiding 312 is $\frac{1}{ts+1}\binom{s(t+1)}{s}$.

This number is also:

- The number of $(t+1)$-ary trees on s nodes
- The number of lattice paths composed of ➔ and 🡅 from (0,0) to (ts, s) that do not cross the line $y=x/t$

Type-β 312-Avoidance



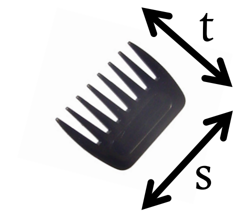
Number of Type-β Comb Extensions Avoiding 312			
s \ t	2	3	4
1	1	1	1
2	3	4	5
3	12	22	35
4	55	140	285
5	273	969	2530
6	1428	7084	23751

Theorem: The number of Type-β comb extensions avoiding 312 is $\frac{1}{ts+1}\binom{s(t+1)}{s}$.

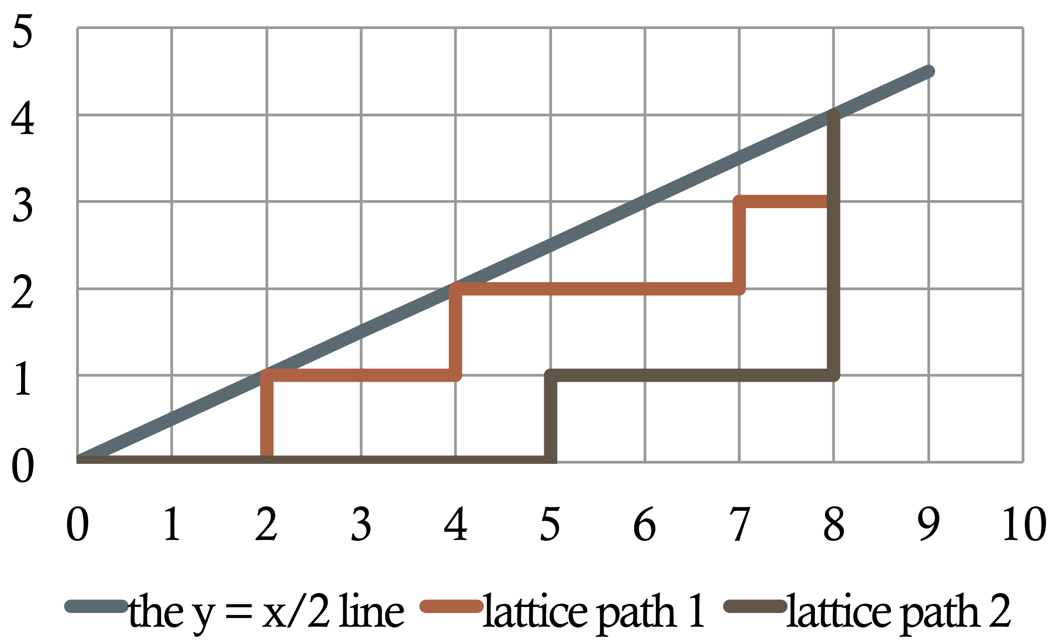
This number is also:

- The number of $(t+1)$-ary trees on s nodes
- The number of lattice paths composed of ➔ and 🡅 from (0,0) to (ts, s) that do not cross the line $y=x/t$
  - ➢ We will show that the number of 312-avoiding type-β comb extensions is equal to the number of such lattice paths

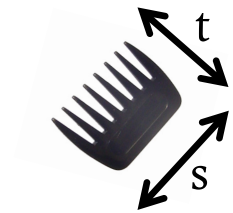
Lattice Paths



lattice paths composed of ➔ and 🡅
from (0,0) to (*ts*, *s*)
that do not cross the line y=x/*t*
(t = 2, s = 4)



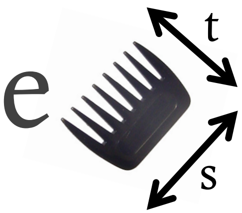
Lattice Paths to Comb Extensions

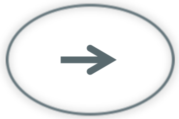


Generating tree analysis shows that # 312-avoiding type-β comb extensions = # lattice paths.

A (somewhat ugly) bijection follows.

Lattice Path Generating Tree

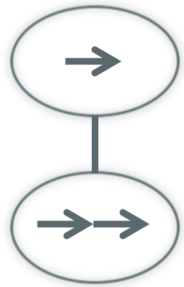




Children are obtained by taking one or more steps up followed by one step to the right

(all possible steps up are assumed to be taken upon reaching x= *st*)

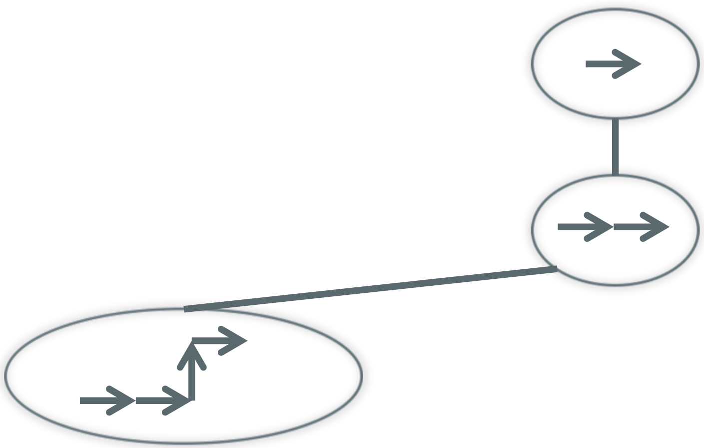
Lattice Path Generating Tree

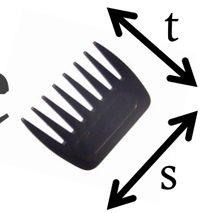


Children are obtained by taking one or more steps up followed by one step to the right

(all possible steps up are assumed to be taken upon reaching x= *st*)

Lattice Path Generating Tree

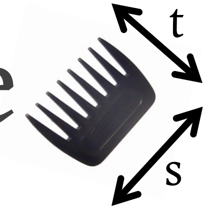


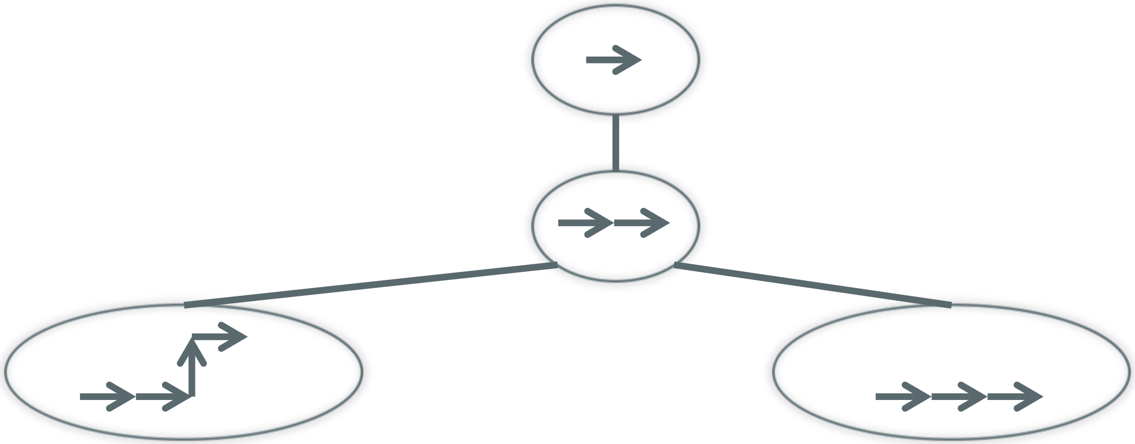


Children are obtained by taking one or more steps up followed by one step to the right

(all possible steps up are assumed to be taken upon reaching x= *st*)

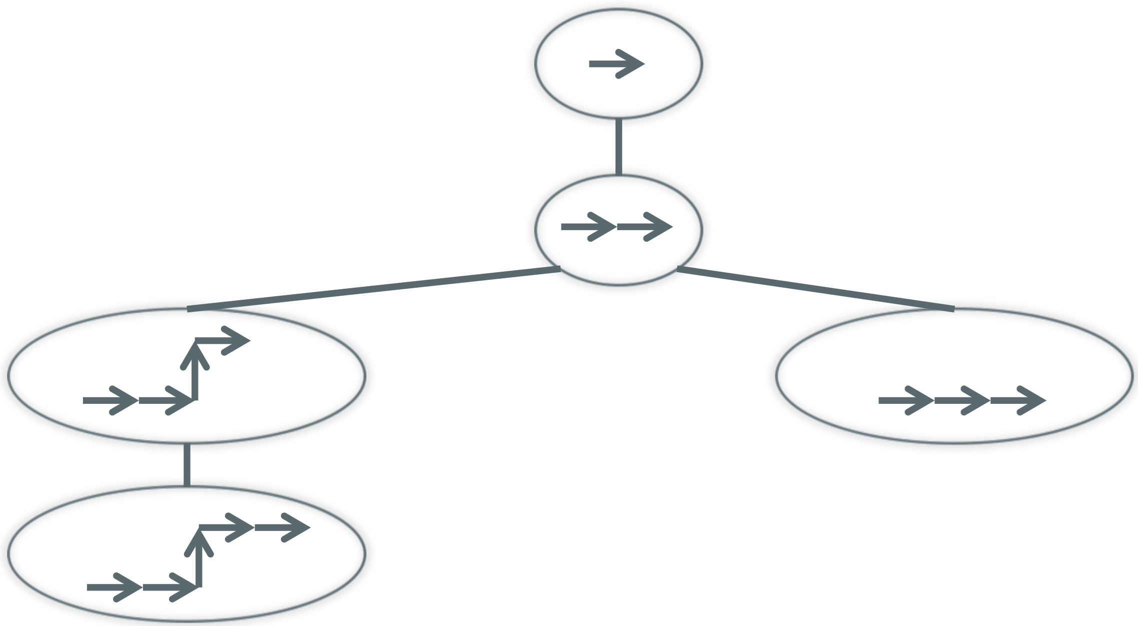
Lattice Path Generating Tree





Children are obtained by taking one or more steps up followed by one step to the right
(all possible steps up are assumed to be taken upon reaching x= *st*)

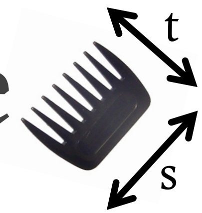
Lattice Path Generating Tree

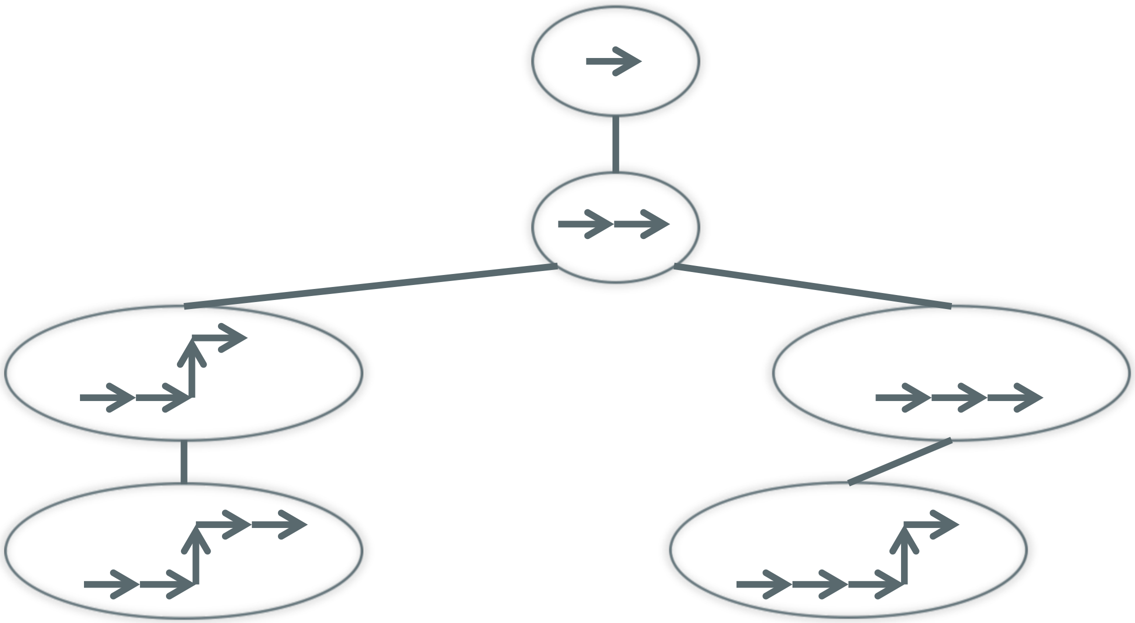


Children are obtained by taking one or more steps up followed by one step to the right

(all possible steps up are assumed to be taken upon reaching x= *st*)

Lattice Path Generating Tree

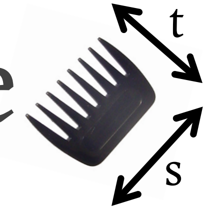


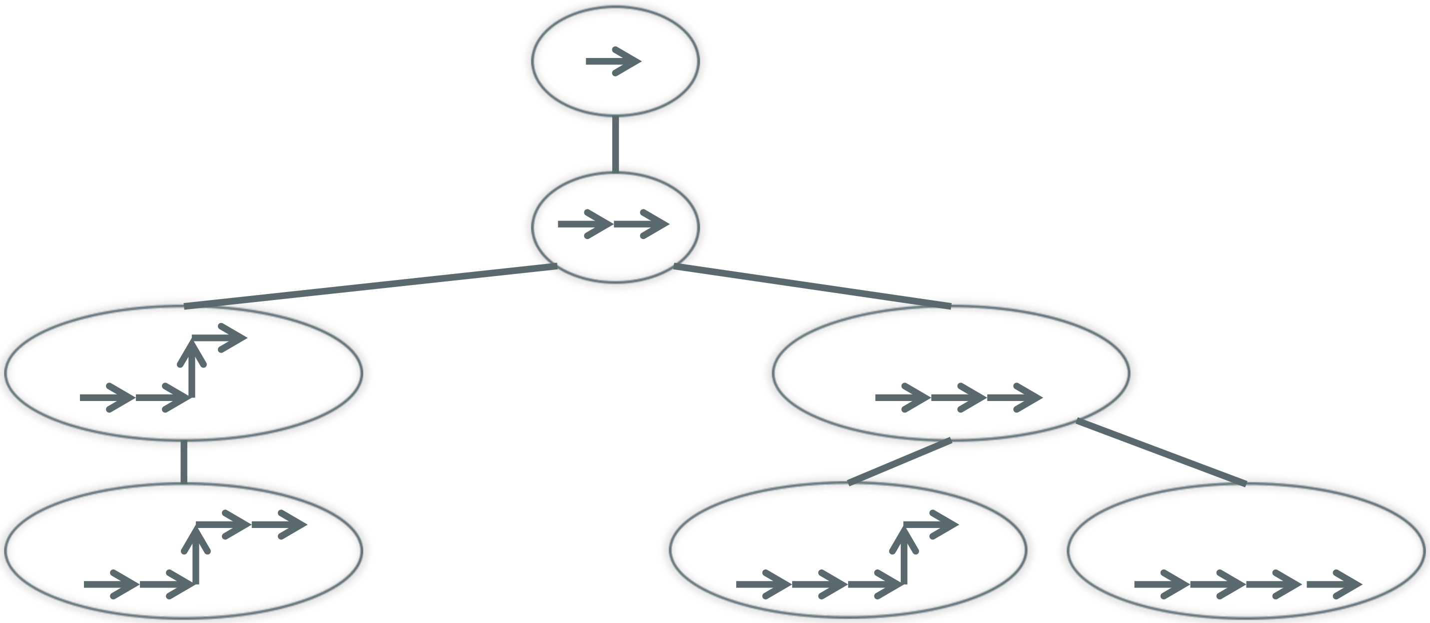


Children are obtained by taking one or more steps up followed by one step to the right

(all possible steps up are assumed to be taken upon reaching x= *st*)

Lattice Path Generating Tree

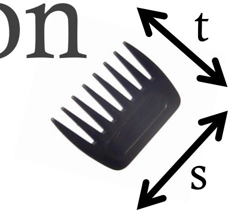


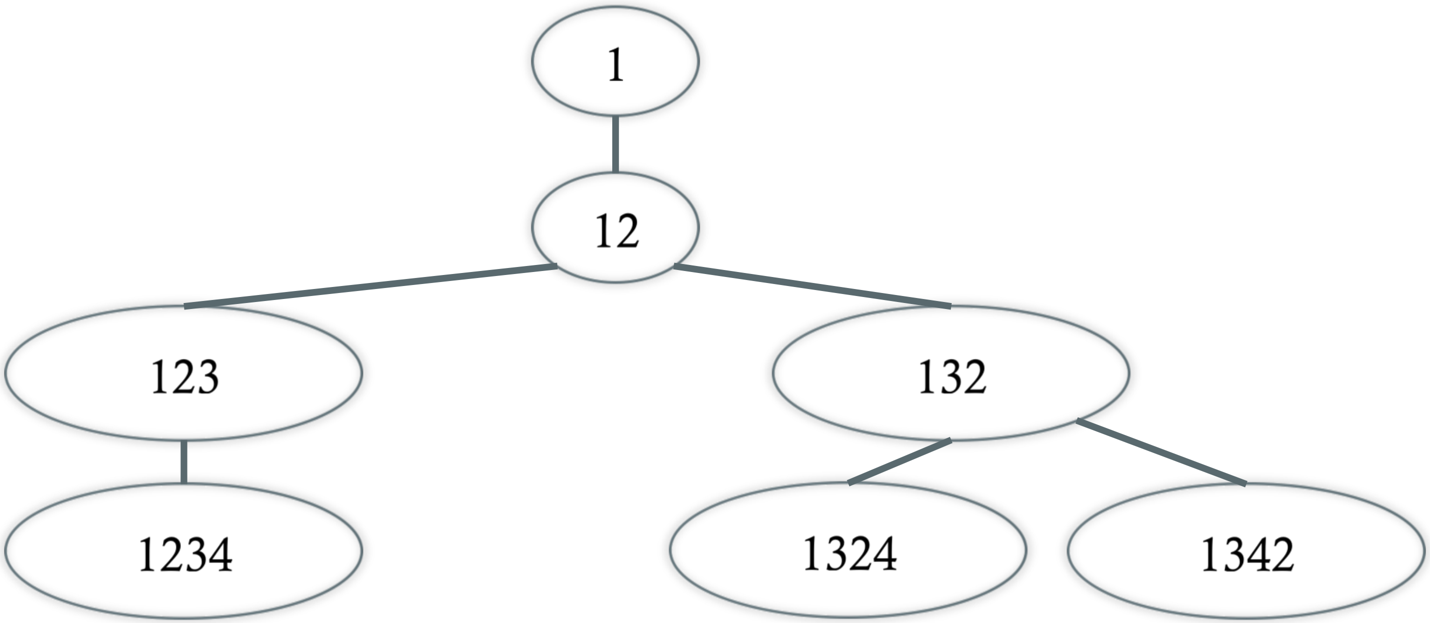


Children are obtained by taking one or more steps up followed by one step to the right

(all possible steps up are assumed to be taken upon reaching x= *st*)

312-Avoiding Comb Extension Generating Tree

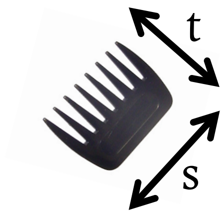




Children are obtained by inserting the next element in such a way as to:

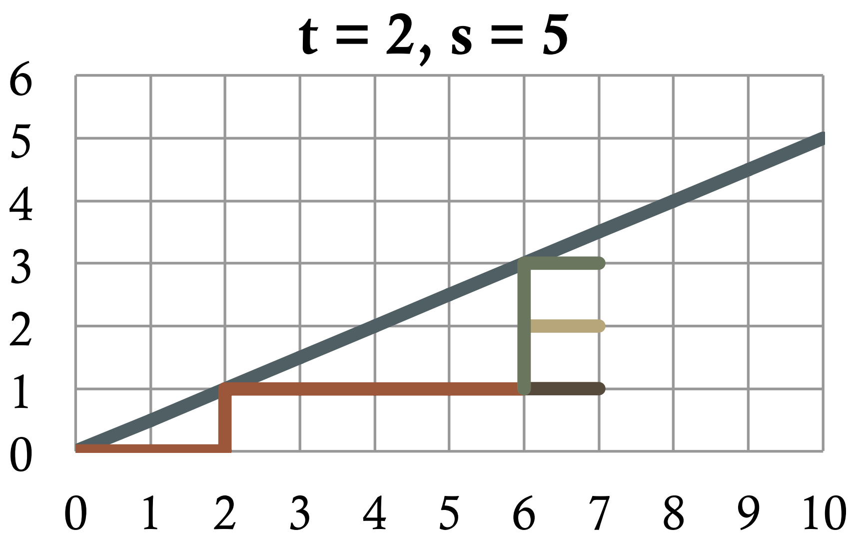
- not violate the comb minimal relations
- not introduce any 312 patterns

Lattice Path Recurrence Relation

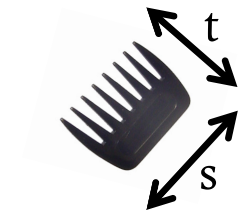


n = size (endpoint x-coordinate) of grandparent path

c = number of children of grandparent path (number of parents)

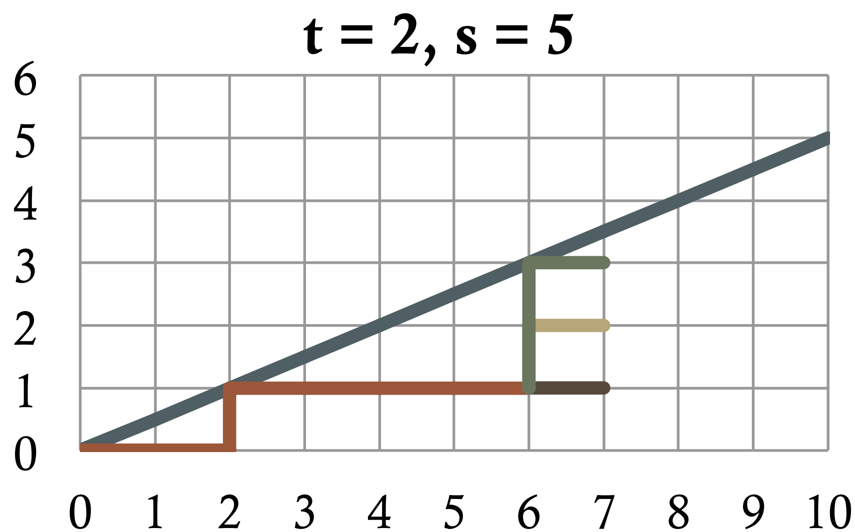


Lattice Path Recurrence Relation

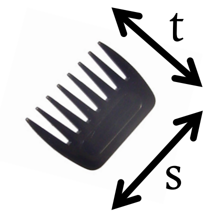


n = size (endpoint x-coordinate) of grandparent path
c = number of children of grandparent path (number of parents)

- If n+1 is not divisible by t: the parent lattice paths will have 1, …, c children



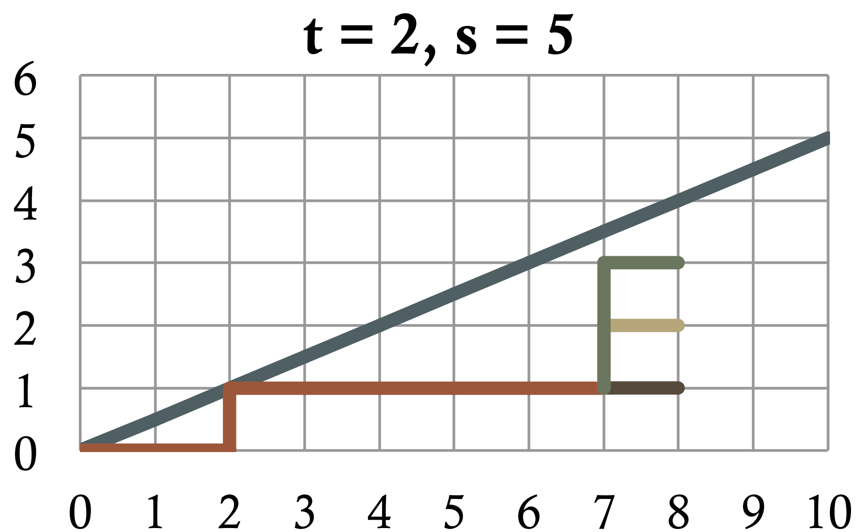
Lattice Path Recurrence Relation



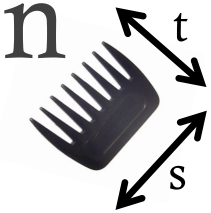
n = size (endpoint x-coordinate) of grandparent path
c = number of children of grandparent path (number of parents)

- If n+1 is not divisible by t: the parent lattice paths will have 1, …, c children

- If n+1 is divisible by t: the parent lattice paths will have 2, …, c+1 children



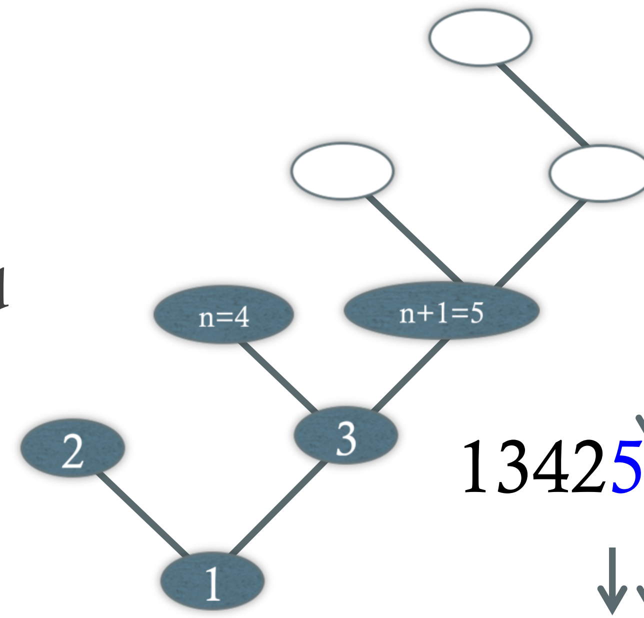
312-Avoiding Comb Extension Recurrence Relation



n = size of grandparent extension
c = number of children of grandparent extension

The next element can always be inserted anywhere *after* a certain point:

- *After* elements that precede it by a minimal relation, and
- *After* the first element of the last 12 pattern

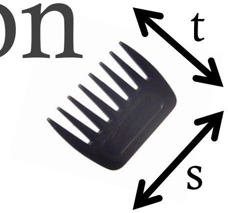


13425

13452

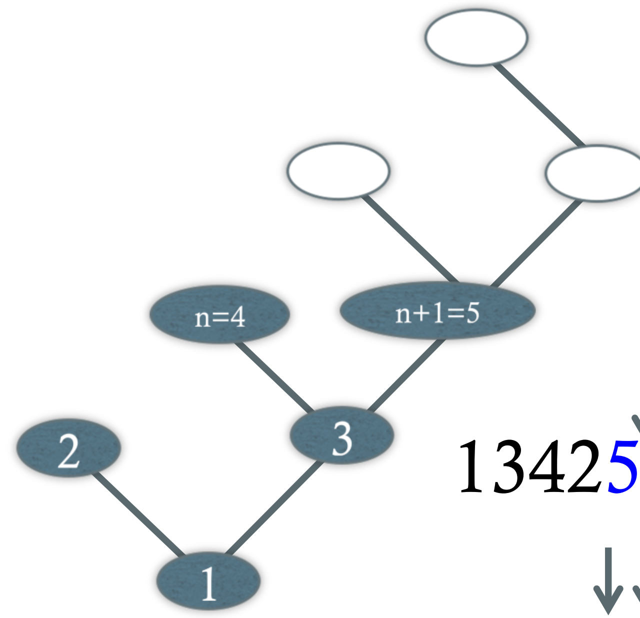
13542

312-Avoiding Comb Extension Recurrence Relation



n = size of grandparent extension
c = number of children of grandparent extension

- If n+1 is not divisible by t:
  - constraint for adding n+2: after the greatest element
  - the n+1 element child comb extensions will have 1, …, c children

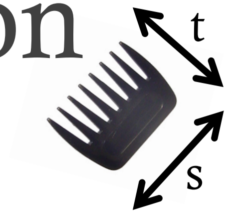


13425

13452

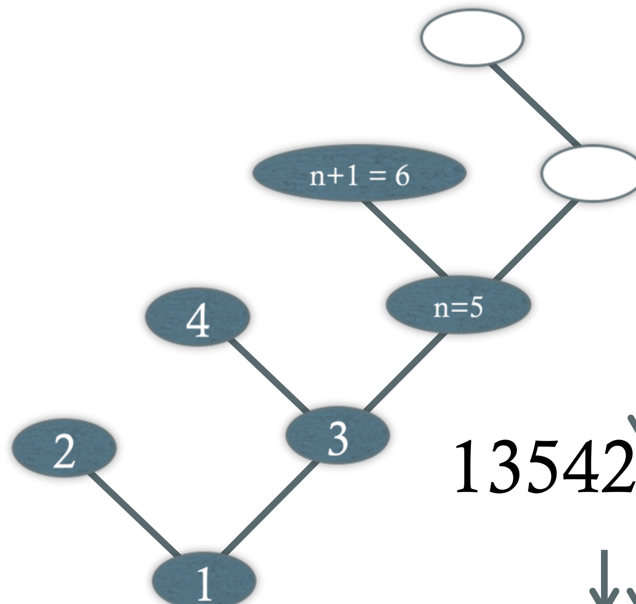
13542

312-Avoiding Comb Extension Recurrence Relation



n = size of grandparent extension
c = number of children of grandparent extension

- If n+1 is divisible by *t*:
  - constraint for adding n+2: immediately before or anywhere after the greatest element
  - the n+1 element child comb extensions will have 2, …, *c*+1 children

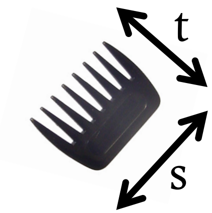


135426

135462

135642

… The Two Side-by-Side



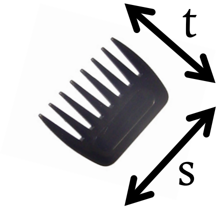
If an n-element comb extension has c children, then:

- If n+1 is not divisible by t: the parent comb extensions will have $1, \ldots, c$ children
- If n+1 is divisible by t: the parent comb extensions will have $2, \ldots, c+1$ children

If a lattice path ending at (n, y) has c children, then:

- If n+1 is not divisible by t: the parent lattice paths will have $1, \ldots, c$ children
- If n+1 is divisible by t: the parent lattice paths will have $2, \ldots, c+1$ children

Proof Complete



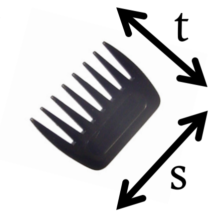
Since both 312-avoiding comb extensions and lattice paths start with:

- a single instance of size 1
- with a single size-2 child

the proof is complete.

$\square$

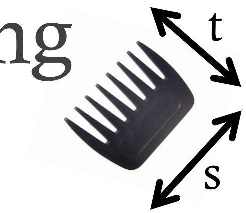
Enumeration of Pattern Avoiding Type-α and β Comb Extensions



Pattern	Number of Avoiding Type-α Comb Extensions	Number of Avoiding Type-β Comb Extensions
123	0	0
132	1	1
213	Catalan Numbers (C_s)	t^{s-1}
231	Open	t^{s-1}
312	$C_{s+1} - C_s$	$\frac{1}{ts+1}\binom{s(t+1)}{s}$
321	Open	Open for $t > 2$

sonka89@mit.edu

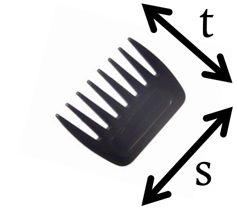
Enumeration of Multi-Pattern Avoiding Type- α and β Comb Extensions



Patterns	Number of Avoiding Type- α Comb Extensions	Number of Avoiding Type-β Comb Extensions
213, 231	2^{s-1}	1
213, 312	2^{s-1}	2^{s-1}
213, 321	$\binom{s}{2} + 1$	$(s-1)(t-1) + 1$
231, 312	Recursive relationship known	2^{s-1}
231, 321	Open for $t > 2$	t^{s-1}
312, 321	Recursive relationship known	$(t+1)^{s-1}$

sonka89@mit.edu

Other loose ends…



- Combs with teeth of varying sizes?
- Other value assignments to the comb elements?
- Longer patterns?

…

- Other (non-comb-like) posets?

sonka89@mit.edu

Aknowledgements

Many thanks to:

- The organizers of this conference, for letting me speak today
- Professor Richard Stanley, for introducing me both to combinatorics and to this problem, and for helping me at every step of the way
- You, for listening!

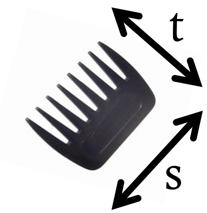
Questions? Suggestions?

sonka89@mit.edu

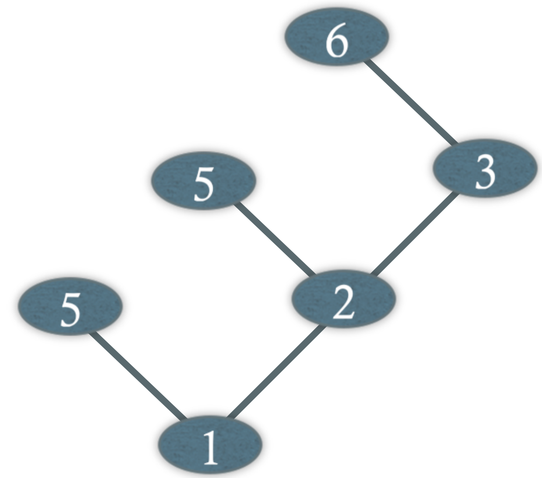
Backup Slides

Just in case…

Type- α Comb Extension Pattern Avoidance Example

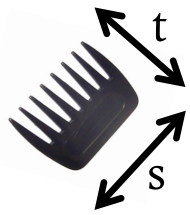


Extension	123	132	213	231	312	321
123456		✔	✔	✔	✔	✔
123465			✔	✔	✔	✔
123546				✔	✔	✔
123564			✔		✔	✔
123645			✔	✔		✔
123654			✔	✔	✔	
124365				✔	✔	✔
124365				✔	✔	✔
124536					✔	✔
125346			✔			✔
125364				✔		✔

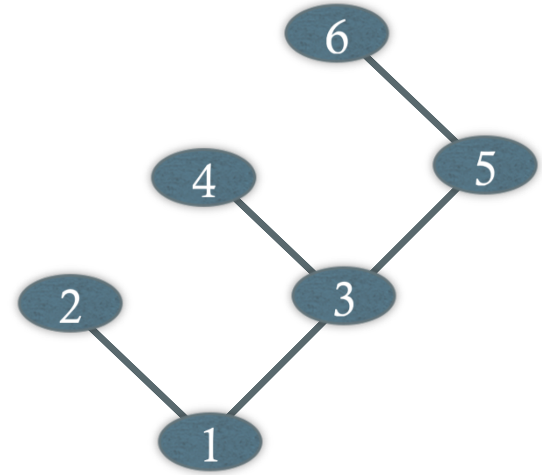


Extension	123	132	213	231	312	321
125436						✔
142356				✔		✔
142365				✔		✔
142536						✔

Type- α Comb Extension Pattern Avoidance Example



Extension	123	132	213	231	312	321
123456		✔	✔	✔	✔	✔
123546				✔	✔	✔
123564			✔		✔	✔
132456				✔	✔	✔
132546				✔	✔	✔
132564					✔	✔
134256					✔	✔
134526					✔	✔
134562			✔		✔	✔
135246						✔
135264						✔



Extension	123	132	213	231	312	321
135426						✔
135462						✔
135624						✔
135642			✔			✔

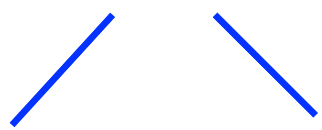
$(t+1)$-ary Tree On s Nodes

$$\frac{1}{ts+1}\binom{s(t+1)}{s}$$

Binary Trees

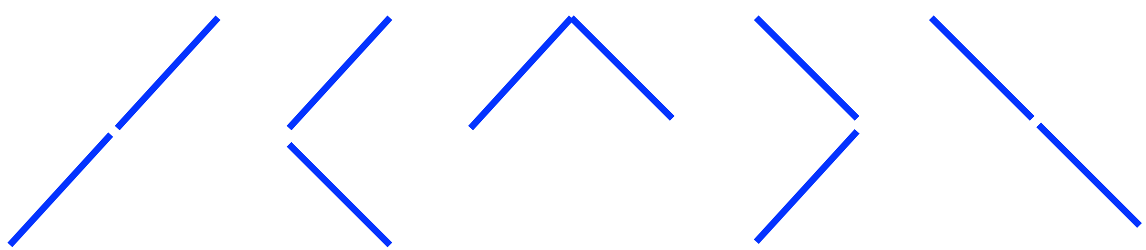
On 2 nodes (t = 1, s = 2):

$$\frac{1}{1\cdot 2+1}\binom{2(1+1)}{2} = \frac{1}{3}\binom{4}{2} = \frac{6}{3} = 2$$



On 3 nodes (t = 1, s = 3):

$$\frac{1}{1\cdot 3+1}\binom{3(1+1)}{3} = \frac{1}{4}\binom{6}{3} = \frac{20}{4} = 5$$

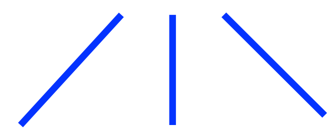


$(t+1)$-ary Tree On s Nodes

$$\frac{1}{ts+1}\binom{s(t+1)}{s}$$

Ternary Trees

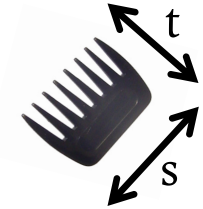
On 2 nodes (t = 2, s = 2): $$\frac{1}{2\cdot 2+1}\binom{2(2+1)}{2} = \frac{1}{5}\binom{6}{2} = \frac{15}{5} = 3$$



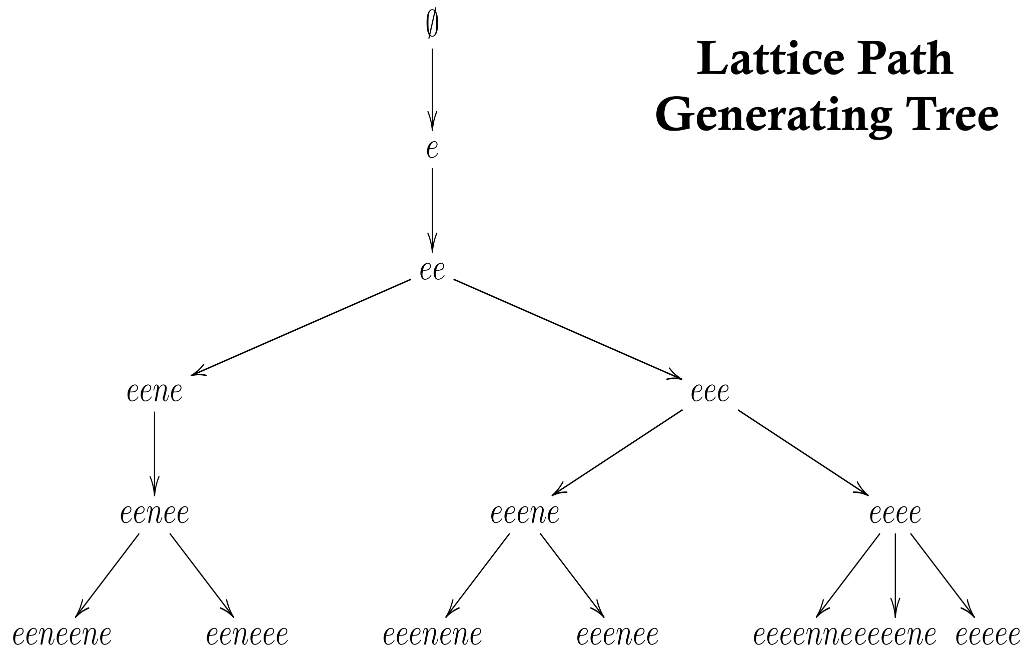
On 3 nodes (t = 2, s = 3): $$\frac{1}{2\cdot 3+1}\binom{3(2+1)}{3} = \frac{1}{7}\binom{9}{3} = \frac{84}{7} = 12$$

Number of ternary trees on n nodes: 1, 3, 12, 55, 273, 1428, …

… The Two Side-by-Side



Lattice Path Generating Tree



312-avoiding Comb Extension Generating Tree