Unlabeled vs labeled objects

Unlabeled

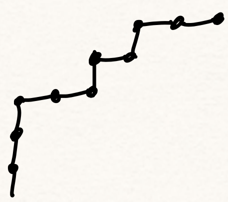
(1) 

3 + 2 + 2

partition

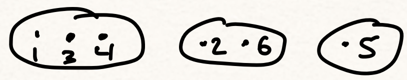
(2)  Binary tree

(3) Dyck path

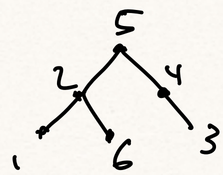


Ordinary generating function

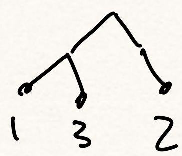
Labeled

(1) 

Set partition

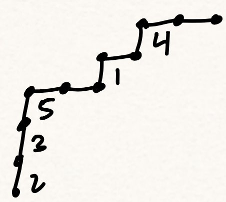
(2) 

labeled binary tree



leaf labeled binary tree

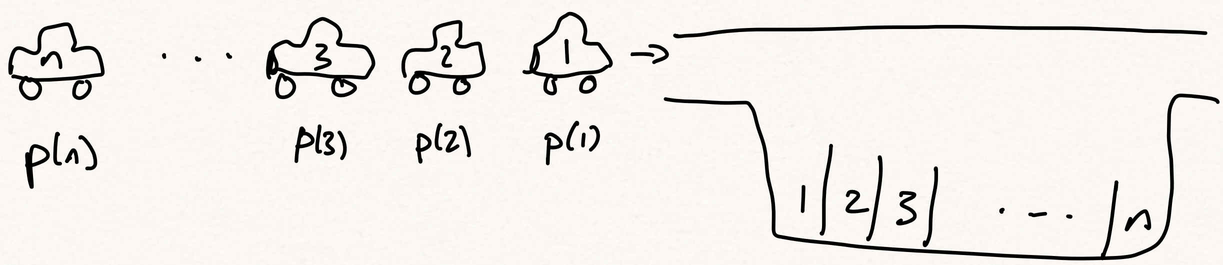
(3) Parking function – label the up-steps $1, \dots, n$ s.t. cols are increasing



Exponential generating function

Parking functions

Q: n cars line up to park in a lot w/ n spots, each car c has a preferred spot $p(c)$:

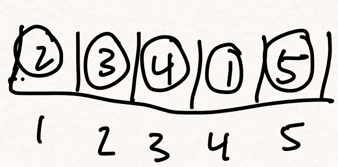


The cars drive to their preferred spot; if it is taken, they park in the next available spot (or exit if no such spot).

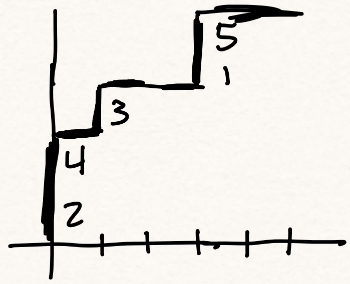
We say p is a <u>parking function</u> if all cars end up parked. Which/how many functions are parking functions?

Ex: $n=2$: $\boxed{\begin{array}{c} p(1)=1 \\ p(2)=2 \end{array}}$ $\boxed{\begin{array}{c} p(1)=2 \\ p(2)=1 \end{array}}$ $\boxed{\begin{array}{c} p(1)=1 \\ p(2)=1 \end{array}}$

Ex: $\left.\begin{array}{l} p(1)=4 \\ p(2)=1 \\ p(3)=2 \\ p(4)=1 \\ p(5)=4 \end{array}\right\}$ write as 41214



Graph: draw car i in col j if $p(i) = j$, stack the cars within col j increasing, have each col start above previous:



Then draw up steps to their left and connect w/ right steps

Claim: p is a parking function iff the path in its graph is a Dyck path.

Pf: For parking space 1 to get filled, there must be a car with $p(1) = 1$.

Then for parking space 2 (and 1) to be filled, there must be at least 2 cars w/ $p(c) \leq 2$

In general we need

$$\left(\#\text{ cars } c \text{ w/ } p(c) \leq i\right) \geq i \quad \text{for all } i.$$

In the path, this means at col i we are at height at least i, meaning it stays weakly above the diagonal.

Thm: # parking functions size n

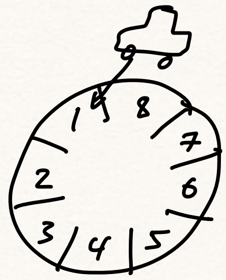
$= (n+1)^{n-1}$

Pf: Add space $n+1$, arrange parking spots in a circle; so all cars park. It's a parking function if $n+1$ is the empty space.

There are $(n+1)^n$ preference functions, and out of $n+1$ ways to rotate, one rotation has $n+1$ as empty space.

So $\frac{1}{n+1}(n+1)^n = (n+1)^{n-1}$

parking functions



QED