

# Probabilités, Statistiques, Combinatoire

Philippe Duchon

Université de Bordeaux – Licence Informatique

2018-2019

## Rappels : codage d'un arbre par mot de Dyck

- ▶ on a vu comment coder un arbre par un mot sur  $\{a, b\}$

## Rappels : codage d'un arbre par mot de Dyck

- ▶ on a vu comment coder un arbre par un mot sur  $\{a, b\}$
- ▶ les mots de codage sont des mots “de Dyck” : la preuve en exercices ; elle repose sur deux propriétés clés, et pas très difficiles :

## Rappels : codage d'un arbre par mot de Dyck

- ▶ on a vu comment coder un arbre par un mot sur  $\{a, b\}$
- ▶ les mots de codage sont des mots “de Dyck” : la preuve en exercices ; elle repose sur deux propriétés clés, et pas très difficiles :
  - ▶ si deux mots  $w_1$  et  $w_2$  sont “équilibrés”, alors  $w = a.w_1.b.w_2$  est équilibré (équilibré : autant de  $a$  que de  $b$ )

## Rappels : codage d'un arbre par mot de Dyck

- ▶ on a vu comment coder un arbre par un mot sur  $\{a, b\}$
- ▶ les mots de codage sont des mots “de Dyck” : la preuve en exercices ; elle repose sur deux propriétés clés, et pas très difficiles :
  - ▶ si deux mots  $w_1$  et  $w_2$  sont “équilibrés”, alors  $w = a.w_1.b.w_2$  est équilibré (équilibré : autant de  $a$  que de  $b$ )
  - ▶ si deux mots  $w_1$  et  $w_2$  sont “positifs”, alors  $w = a.w_1.b.w_2$  est positif (positif : les préfixes ont au moins autant de  $a$  que de  $b$ )

## Rappels : codage d'un arbre par mot de Dyck

- ▶ on a vu comment coder un arbre par un mot sur  $\{a, b\}$
- ▶ les mots de codage sont des mots “de Dyck” : la preuve en exercices ; elle repose sur deux propriétés clés, et pas très difficiles :
  - ▶ si deux mots  $w_1$  et  $w_2$  sont “équilibrés”, alors  $w = a.w_1.b.w_2$  est équilibré (équilibré : autant de  $a$  que de  $b$ )
  - ▶ si deux mots  $w_1$  et  $w_2$  sont “positifs”, alors  $w = a.w_1.b.w_2$  est positif (positif : les préfixes ont au moins autant de  $a$  que de  $b$ )
- ▶ un peu moins évident : deux arbres distincts sont toujours codés par des mots distincts
  - ▶ si  $w_1, w'_1, w_2, w'_2$  sont des mots quelconques, alors **on peut** avoir  $a.w_1.b.w_2 = a.w'_1.b.w'_2$  sans que  $w_1 = w'_1$  et  $w_2 = w'_2$ ...
  - ▶ **mais** si on sait que  $w_1, w'_1, w_2$  et  $w'_2$  sont des mots de Dyck, alors on peut déduire  $w_1 = w'_1$  et  $w_2 = w'_2$  de  $a.w_1.b.w_2 = a.w'_1.b.w'_2$

## Rappels : codage d'un arbre par mot de Dyck

- ▶ on a vu comment coder un arbre par un mot sur  $\{a, b\}$
- ▶ les mots de codage sont des mots “de Dyck” : la preuve en exercices ; elle repose sur deux propriétés clés, et pas très difficiles :
  - ▶ si deux mots  $w_1$  et  $w_2$  sont “équilibrés”, alors  $w = a.w_1.b.w_2$  est équilibré (équilibré : autant de  $a$  que de  $b$ )
  - ▶ si deux mots  $w_1$  et  $w_2$  sont “positifs”, alors  $w = a.w_1.b.w_2$  est positif (positif : les préfixes ont au moins autant de  $a$  que de  $b$ )
- ▶ un peu moins évident : deux arbres distincts sont toujours codés par des mots distincts
  - ▶ si  $w_1, w'_1, w_2, w'_2$  sont des mots quelconques, alors **on peut** avoir  $a.w_1.b.w_2 = a.w'_1.b.w'_2$  sans que  $w_1 = w'_1$  et  $w_2 = w'_2$ ...
  - ▶ **mais** si on sait que  $w_1, w'_1, w_2$  et  $w'_2$  sont des mots de Dyck, alors on peut déduire  $w_1 = w'_1$  et  $w_2 = w'_2$  de  $a.w_1.b.w_2 = a.w'_1.b.w'_2$  (pourquoi ?)

## Rappels : codage d'un arbre par mot de Dyck

- ▶ on a vu comment coder un arbre par un mot sur  $\{a, b\}$
- ▶ les mots de codage sont des mots “de Dyck” : la preuve en exercices ; elle repose sur deux propriétés clés, et pas très difficiles :
  - ▶ si deux mots  $w_1$  et  $w_2$  sont “équilibrés”, alors  $w = a.w_1.b.w_2$  est équilibré (équilibré : autant de  $a$  que de  $b$ )
  - ▶ si deux mots  $w_1$  et  $w_2$  sont “positifs”, alors  $w = a.w_1.b.w_2$  est positif (positif : les préfixes ont au moins autant de  $a$  que de  $b$ )
- ▶ un peu moins évident : deux arbres distincts sont toujours codés par des mots distincts
  - ▶ si  $w_1, w'_1, w_2, w'_2$  sont des mots quelconques, alors **on peut** avoir  $a.w_1.b.w_2 = a.w'_1.b.w'_2$  sans que  $w_1 = w'_1$  et  $w_2 = w'_2 \dots$
  - ▶ **mais** si on sait que  $w_1, w'_1, w_2$  et  $w'_2$  sont des mots de Dyck, alors on peut déduire  $w_1 = w'_1$  et  $w_2 = w'_2$  de  $a.w_1.b.w_2 = a.w'_1.b.w'_2$  (pourquoi ?)
- ▶ enfin : tout mot de Dyck est le codage d'un (unique) arbre

# Décodage d'un mot de Dyck vers un arbre

- ▶ La difficulté dans le décodage d'un mot/chemin consiste à trouver “où le couper” pour trouver où se termine le codage du sous-arbre gauche.

# Décodage d'un mot de Dyck vers un arbre

- ▶ La difficulté dans le décodage d'un mot/chemin consiste à trouver “où le couper” pour trouver où se termine le codage du sous-arbre gauche.
- ▶ En fait on ne peut couper qu'en un seul endroit : **la partie  $a.w_1.b$  correspond forcément au premier préfixe  $u$  non vide pour lequel on a  $|u|_a = |u|_b$**

# Décodage d'un mot de Dyck vers un arbre

- ▶ La difficulté dans le décodage d'un mot/chemin consiste à trouver “où le couper” pour trouver où se termine le codage du sous-arbre gauche.
- ▶ En fait on ne peut couper qu'en un seul endroit : **la partie  $a.w_1.b$  correspond forcément au premier préfixe  $u$  non vide pour lequel on a  $|u|_a = |u|_b$**
- ▶ La partie  $v$  qui suit est aussi forcément un mot de Dyck

# Décodage d'un mot de Dyck vers un arbre

- ▶ La difficulté dans le décodage d'un mot/chemin consiste à trouver “où le couper” pour trouver où se termine le codage du sous-arbre gauche.
- ▶ En fait on ne peut couper qu'en un seul endroit : **la partie  $a.w_1.b$  correspond forcément au premier préfixe  $u$  non vide pour lequel on a  $|u|_a = |u|_b$**
- ▶ La partie  $v$  qui suit est aussi forcément un mot de Dyck
- ▶ Donc un mot de Dyck est le codage d'un et d'un seul arbre binaire : le codage est bien bijectif.

## Comptage des mots de Dyck

On a prouvé qu'il y a autant de mots de Dyck (de longueur  $2n$ ) que d'arbres binaires complets à  $n$  nœuds internes ; mais on n'a pas encore trouvé de formule pour  $c_n$ .

# Comptage des mots de Dyck

On a prouvé qu'il y a autant de mots de Dyck (de longueur  $2n$ ) que d'arbres binaires complets à  $n$  nœuds internes ; mais on n'a pas encore trouvé de formule pour  $c_n$ .

## Théorème

Le nombre de chemins de Dyck de longueur  $2n$ , qui est aussi le nombre de mots de Dyck de longueur  $2n$ , qui est aussi le nombre d'arbres binaires complets ayant  $n$  nœuds internes, vaut

$$c_n = \frac{1}{n+1} \binom{2n}{n} = \frac{(2n)!}{n!(n+1)!}.$$

# Comptage des mots de Dyck

On a prouvé qu'il y a autant de mots de Dyck (de longueur  $2n$ ) que d'arbres binaires complets à  $n$  nœuds internes ; mais on n'a pas encore trouvé de formule pour  $c_n$ .

## Théorème

Le nombre de chemins de Dyck de longueur  $2n$ , qui est aussi le nombre de mots de Dyck de longueur  $2n$ , qui est aussi le nombre d'arbres binaires complets ayant  $n$  nœuds internes, vaut

$$c_n = \frac{1}{n+1} \binom{2n}{n} = \frac{(2n)!}{n!(n+1)!}.$$

**Preuve** : on va prouver par bijection la relation,  $(n+1)c_n = \binom{2n}{n}$ .

# Interprétation combinatoire de $\binom{2n}{n}$

$\binom{2n}{n}$ , c'est le nombre de

- ▶ parties de taille  $n$  de l'ensemble  $[[1, 2n]]$

# Interprétation combinatoire de $\binom{2n}{n}$

$\binom{2n}{n}$ , c'est le nombre de

- ▶ parties de taille  $n$  de l'ensemble  $[[1, 2n]]$
- ▶ mots sur l'alphabet  $\{a, b\}$  tels que  $|w|_a = |w|_b = n$

# Interprétation combinatoire de $\binom{2n}{n}$

$\binom{2n}{n}$ , c'est le nombre de

- ▶ parties de taille  $n$  de l'ensemble  $[[1, 2n]]$
- ▶ mots sur l'alphabet  $\{a, b\}$  tels que  $|w|_a = |w|_b = n$
- ▶ chemins à pas Nord-Est et Sud-Est, partant de  $(0, 0)$ , et terminant en  $(2n, 0)$ , mais sans la condition de positivité des chemins de Dyck.

# Interprétation combinatoire de $\binom{2n}{n}$

$\binom{2n}{n}$ , c'est le nombre de

- ▶ parties de taille  $n$  de l'ensemble  $[[1, 2n]]$
- ▶ mots sur l'alphabet  $\{a, b\}$  tels que  $|w|_a = |w|_b = n$
- ▶ chemins à pas Nord-Est et Sud-Est, partant de  $(0, 0)$ , et terminant en  $(2n, 0)$ , mais sans la condition de positivité des chemins de Dyck. (Pourquoi?)

# Interprétation combinatoire de $\binom{2n}{n}$

$\binom{2n}{n}$ , c'est le nombre de

- ▶ parties de taille  $n$  de l'ensemble  $[[1, 2n]]$
- ▶ mots sur l'alphabet  $\{a, b\}$  tels que  $|w|_a = |w|_b = n$
- ▶ chemins à pas Nord-Est et Sud-Est, partant de  $(0, 0)$ , et terminant en  $(2n, 0)$ , mais sans la condition de positivité des chemins de Dyck. (Pourquoi?)

On cherche donc une bijection entre

- ▶ les chemins de  $(0, 0)$  à  $(2n, 0)$ , à pas NE/SE, sans condition de positivité, d'une part ; et
- ▶ les couples formés d'un entier pris dans  $[[1, n + 1]]$  d'une part, et d'un chemin de Dyck de  $(0, 0)$  à  $(2n, 0)$ , d'autre part.

## Transformation : “lemme cyclique”

On part d'un chemin/mot équilibré  $w$

- ▶ On ajoute un pas Sud-Est à la fin :  $w' = w.b$  (chemin se terminant en  $(2n + 1, -1)$ , par un pas Sud-Est)

## Transformation : “lemme cyclique”

On part d'un chemin/mot équilibré  $w$

- ▶ On ajoute un pas Sud-Est à la fin :  $w' = w.b$  (chemin se terminant en  $(2n + 1, -1)$ , par un pas Sud-Est)
- ▶ On remarque que  $w'$  a exactement  $n + 1$  pas Sud-Est (et  $n$  pas Nord-Est)

## Transformation : “lemme cyclique”

On part d'un chemin/mot équilibré  $w$

- ▶ On ajoute un pas Sud-Est à la fin :  $w' = w.b$  (chemin se terminant en  $(2n + 1, -1)$ , par un pas Sud-Est)
- ▶ On remarque que  $w'$  a exactement  $n + 1$  pas Sud-Est (et  $n$  pas Nord-Est)
- ▶ On repère le premier point où  $w'$  atteint sa hauteur minimale (forcément négative) : c'est forcément avec un pas Sud-Est,  $w' = u.b.v.b$  (le point est après  $u.b$ ) (cas particulier : on peut avoir  $u.b = w'$ , la seconde partie est alors vide)

## Transformation : “lemme cyclique”

On part d'un chemin/mot équilibré  $w$

- ▶ On ajoute un pas Sud-Est à la fin :  $w' = w.b$  (chemin se terminant en  $(2n + 1, -1)$ , par un pas Sud-Est)
- ▶ On remarque que  $w'$  a exactement  $n + 1$  pas Sud-Est (et  $n$  pas Nord-Est)
- ▶ On repère le premier point où  $w'$  atteint sa hauteur minimale (forcément négative) : c'est forcément avec un pas Sud-Est,  $w' = u.b.v.b$  (le point est après  $u.b$ ) (cas particulier : on peut avoir  $u.b = w'$ , la seconde partie est alors vide)
- ▶ On note  $k = |ub|_b = 1 + |u|_b \in [[1, n + 1]]$  (le cas particulier correspond à  $k = n + 1$ )

## Transformation : “lemme cyclique”

On part d'un chemin/mot équilibré  $w$

- ▶ On ajoute un pas Sud-Est à la fin :  $w' = w.b$  (chemin se terminant en  $(2n + 1, -1)$ , par un pas Sud-Est)
- ▶ On remarque que  $w'$  a exactement  $n + 1$  pas Sud-Est (et  $n$  pas Nord-Est)
- ▶ On repère le premier point où  $w'$  atteint sa hauteur minimale (forcément négative) : c'est forcément avec un pas Sud-Est,  $w' = u.b.v.b$  (le point est après  $u.b$ ) (cas particulier : on peut avoir  $u.b = w'$ , la seconde partie est alors vide)
- ▶ On note  $k = |ub|_b = 1 + |u|_b \in [[1, n + 1]]$  (le cas particulier correspond à  $k = n + 1$ )
- ▶ On échange les parties  $u.b$  et  $v.b$ , et on élimine le  $b$  final ; *i.e.* on forme le mot  $w'' = v.b.u$  : on a perdu un  $b$  donc  $|w''|_a = |w''|_b = n$ .

## Transformation : “lemme cyclique”

On part d'un chemin/mot équilibré  $w$

- ▶ On ajoute un pas Sud-Est à la fin :  $w' = w.b$  (chemin se terminant en  $(2n + 1, -1)$ , par un pas Sud-Est)
- ▶ On remarque que  $w'$  a exactement  $n + 1$  pas Sud-Est (et  $n$  pas Nord-Est)
- ▶ On repère le premier point où  $w'$  atteint sa hauteur minimale (forcément négative) : c'est forcément avec un pas Sud-Est,  $w' = u.b.v.b$  (le point est après  $u.b$ ) (cas particulier : on peut avoir  $u.b = w'$ , la seconde partie est alors vide)
- ▶ On note  $k = |ub|_b = 1 + |u|_b \in [[1, n + 1]]$  (le cas particulier correspond à  $k = n + 1$ )
- ▶ On échange les parties  $u.b$  et  $v.b$ , et on élimine le  $b$  final ; *i.e.* on forme le mot  $w'' = v.b.u$  : on a perdu un  $b$  donc  $|w''|_a = |w''|_b = n$ .
- ▶ **Lemme** : le mot  $w''$  est un mot de Dyck.

## Transformation : “lemme cyclique”

On part d'un chemin/mot équilibré  $w$

- ▶ On ajoute un pas Sud-Est à la fin :  $w' = w.b$  (chemin se terminant en  $(2n + 1, -1)$ , par un pas Sud-Est)
- ▶ On remarque que  $w'$  a exactement  $n + 1$  pas Sud-Est (et  $n$  pas Nord-Est)
- ▶ On repère le premier point où  $w'$  atteint sa hauteur minimale (forcément négative) : c'est forcément avec un pas Sud-Est,  $w' = u.b.v.b$  (le point est après  $u.b$ ) (cas particulier : on peut avoir  $u.b = w'$ , la seconde partie est alors vide)
- ▶ On note  $k = |ub|_b = 1 + |u|_b \in [[1, n + 1]]$  (le cas particulier correspond à  $k = n + 1$ )
- ▶ On échange les parties  $u.b$  et  $v.b$ , et on élimine le  $b$  final ; *i.e.* on forme le mot  $w'' = v.b.u$  : on a perdu un  $b$  donc  $|w''|_a = |w''|_b = n$ .
- ▶ **Lemme** : le mot  $w''$  est un mot de Dyck.
- ▶ Le résultat de la transformation  $\varphi$  en partant de  $w$ , est le couple  $(k, w'')$ .

# Transformation inverse

Pour s'assurer qu'il s'agit bien d'une bijection, il faut montrer deux choses, pour n'importe quel couple  $(k, w'')$  :

- ▶ on peut “revenir en arrière”, et trouver un  $w$  tel que  $\varphi(w) = (k, w'')$  (la transformation est **surjective**)
- ▶ on n'a pas le choix pour trouver ce  $w$ , il n'y en a qu'un (la transformation est **injective**)

Partant de  $w''$ , il faut trouver “où couper” le mot  $w''b$  pour former  $vb$  et  $ub$ ; mais on a  $k$  qui nous y aide :  $|ub|_b = k$ , donc  $|u|_b = k - 1$  : le  $b$  final de  $vb$  est donc le  $k$ -ème  $b$  de  $w''b$  en partant de la fin (et on n'a pas le choix, c'est lui qu'il faut prendre). (Dans le cas particulier  $k = n + 1$ ,  $u.b = w''.b$  et la partie “ $v.b$ ” est le mot vide)

(Une fois qu'on a identifié  $u$  et  $v$ , on retrouve  $w = ubv$ )

# Le lemme cyclique sur un exemple

On part d'un mot  $w$  :

$a \ b \ b \ a \ b \ b \ a \ b \ a \ a \ b \ a \ a \ b$

## Le lemme cyclique sur un exemple

On part d'un mot  $w$  :

$a$	$b$	$b$	$a$	$b$	$b$	$a$	$b$	$a$	$a$	$b$	$a$	$a$	$b$	$b$
1	0	-1	0	-1	-2	-1	-2	-1	0	-1	0	1	0	-1

## Le lemme cyclique sur un exemple

On part d'un mot  $w$  :

$a$	$b$	$b$	$a$	$b$	$b$	$a$	$b$	$a$	$a$	$b$	$a$	$a$	$b$	$b$
1	0	-1	0	-1	-2	-1	-2	-1	0	-1	0	1	0	-1

On obtient  $ub = abbabb$ ,  $vb = abaabaabb$ ,  $k = 4$ .

## Le lemme cyclique sur un exemple

On part d'un mot  $w$  :

$a$	$b$	$b$	$a$	$b$	$b$	$a$	$b$	$a$	$a$	$b$	$a$	$a$	$b$	$b$
1	0	-1	0	-1	-2	-1	-2	-1	0	-1	0	1	0	-1

On obtient  $ub = abbabb$ ,  $vb = abaabaabb$ ,  $k = 4$ .

On obtient donc  $w'' = abaabaabbabbab$

## Lemme cyclique : exemple dans l'autre sens

- ▶ On prend  $w'' = abaabaabbabbab$ ,  $k = 4$

## Lemme cyclique : exemple dans l'autre sens

- ▶ On prend  $w'' = abaabaabbabbab$ ,  $k = 4$
- ▶ On coupe après le 4ème  $b$  en partant de la fin du mot :

$$w'' = abaabaabb \cdot abbab$$

## Lemme cyclique : exemple dans l'autre sens

- ▶ On prend  $w'' = abaabaabbabbab$ ,  $k = 4$
- ▶ On coupe après le 4ème  $b$  en partant de la fin du mot :

$$w'' = \text{abaabaabb} \cdot \text{abbab}$$

- ▶ On ajoute un  $b$  dans la partie droite, on échange les deux parties, puis on efface le nouveau  $b$  final :

$$w = \text{abbabb} \cdot \text{abaabaab}$$

(cas particulier : si  $k = n + 1$ , on retrouve  $w = w''$ )

## Lemme cyclique : exemple dans l'autre sens

- ▶ On prend  $w'' = abaabaabbabbab$ ,  $k = 4$
- ▶ On coupe après le 4ème  $b$  en partant de la fin du mot :

$$w'' = \text{abaabaabb} \cdot \text{abbab}$$

- ▶ On ajoute un  $b$  dans la partie droite, on échange les deux parties, puis on efface le nouveau  $b$  final :

$$w = \text{abbabb} \cdot \text{abaabaab}$$

(cas particulier : si  $k = n + 1$ , on retrouve  $w = w''$ )

- ▶ (On a bien retrouvé le mot  $w$  de départ)

## Une autre preuve : “lemme de réflexion”

(“réflexion” au sens d’image miroir, pas de penser intensivement, encore que...)

Il existe une autre preuve classique de la formule pour les nombres de Catalan, basée non pas sur le *produit*  $(n+1)c_n = \binom{2n}{n}$ , mais sur une *différence* :

$$c_n = \binom{2n}{n} - \binom{2n}{n-1}$$

(formule qui se prouve par manipulation de factorielles)

On réécrit l’identité comme une somme :

$$\binom{2n}{n} = c_n + \binom{2n}{n-1}$$

et on montre une correspondance bijective entre deux ensembles :

- ▶ les chemins (Nord-Est, Sud-Est) de  $(0,0)$  à  $(2n,0)$ , d’une part ;
- ▶ les chemins qui sont soit des chemins de Dyck (de  $(0,0)$  à  $(2n,n)$ ), soit des chemins (Nord-Est, Sud-Est) de  $(0,0)$  à  $(2n,-2)$ , d’autre part.

## Les nombres de Catalan sont partout. . .

- ▶ La même séquence  $c_n = \frac{1}{n+1} \binom{2n}{n}$  compte à la fois les mots de Dyck (de longueur  $2n$ ) et les arbres binaires complets (à  $n$  nœuds internes).

## Les nombres de Catalan sont partout. . .

- ▶ La même séquence  $c_n = \frac{1}{n+1} \binom{2n}{n}$  compte à la fois les mots de Dyck (de longueur  $2n$ ) et les arbres binaires complets (à  $n$  nœuds internes).
- ▶ En fait, cette suite compte également pléthore d'autres classes combinatoires! (**OEIS**, A000108), pour lesquelles on trouve souvent soit une bijection "facile" avec les mots de Dyck ou les arbres binaires, soit la récurrence trouvée pour les arbres.

## Les nombres de Catalan sont partout. . .

- ▶ La même séquence  $c_n = \frac{1}{n+1} \binom{2n}{n}$  compte à la fois les mots de Dyck (de longueur  $2n$ ) et les arbres binaires complets (à  $n$  nœuds internes).
- ▶ En fait, cette suite compte également pléthore d'autres classes combinatoires! (**OEIS**, A000108), pour lesquelles on trouve souvent soit une bijection "facile" avec les mots de Dyck ou les arbres binaires, soit la récurrence trouvée pour les arbres.
- ▶ arbres binaires, complets ou non, à  $n$  nœuds

## Les nombres de Catalan sont partout. . .

- ▶ La même séquence  $c_n = \frac{1}{n+1} \binom{2n}{n}$  compte à la fois les mots de Dyck (de longueur  $2n$ ) et les arbres binaires complets (à  $n$  nœuds internes).
- ▶ En fait, cette suite compte également pléthore d'autres classes combinatoires! (**OEIS**, A000108), pour lesquelles on trouve souvent soit une bijection "facile" avec les mots de Dyck ou les arbres binaires, soit la récurrence trouvée pour les arbres.
- ▶ arbres binaires, complets ou non, à  $n$  nœuds
- ▶ arbres plans non vides à  $n + 1$  nœuds (arbres plans : chaque nœud peut avoir un nombre arbitraire de fils ; les fils, s'il y en a plusieurs, sont ordonnés arbitrairement)

## Les nombres de Catalan sont partout. . .

- ▶ La même séquence  $c_n = \frac{1}{n+1} \binom{2n}{n}$  compte à la fois les mots de Dyck (de longueur  $2n$ ) et les arbres binaires complets (à  $n$  nœuds internes).
- ▶ En fait, cette suite compte également pléthore d'autres classes combinatoires! (**OEIS**, A000108), pour lesquelles on trouve souvent soit une bijection "facile" avec les mots de Dyck ou les arbres binaires, soit la récurrence trouvée pour les arbres.
- ▶ arbres binaires, complets ou non, à  $n$  nœuds
- ▶ arbres plans non vides à  $n + 1$  nœuds (arbres plans : chaque nœud peut avoir un nombre arbitraire de fils ; les fils, s'il y en a plusieurs, sont ordonnés arbitrairement)
- ▶ façons de tracer  $n - 1$  cordes dans un polygone régulier à  $n + 2$  côtés, sans qu'elles se croisent autrement qu'aux sommets

## Les nombres de Catalan sont partout. . .

- ▶ La même séquence  $c_n = \frac{1}{n+1} \binom{2n}{n}$  compte à la fois les mots de Dyck (de longueur  $2n$ ) et les arbres binaires complets (à  $n$  nœuds internes).
- ▶ En fait, cette suite compte également pléthore d'autres classes combinatoires! (**OEIS**, A000108), pour lesquelles on trouve souvent soit une bijection "facile" avec les mots de Dyck ou les arbres binaires, soit la récurrence trouvée pour les arbres.
- ▶ arbres binaires, complets ou non, à  $n$  nœuds
- ▶ arbres plans non vides à  $n + 1$  nœuds (arbres plans : chaque nœud peut avoir un nombre arbitraire de fils ; les fils, s'il y en a plusieurs, sont ordonnés arbitrairement)
- ▶ façons de tracer  $n - 1$  cordes dans un polygone régulier à  $n + 2$  côtés, sans qu'elles se croisent autrement qu'aux sommets
- ▶ façons de relier deux à deux  $2n$  points sur un cercle, sans croisements