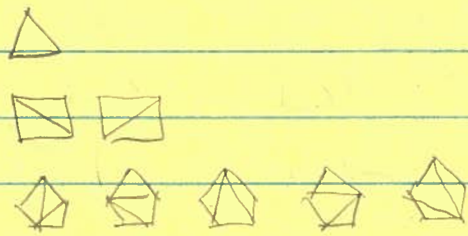
L15 Other interpretations of Catalan Numbers.

§2.6.6 HHM Q How many ways can one divide an $(n+2)$-gon into triangles?

A.



	n	#ways
	1	1
	2	2
	3	5

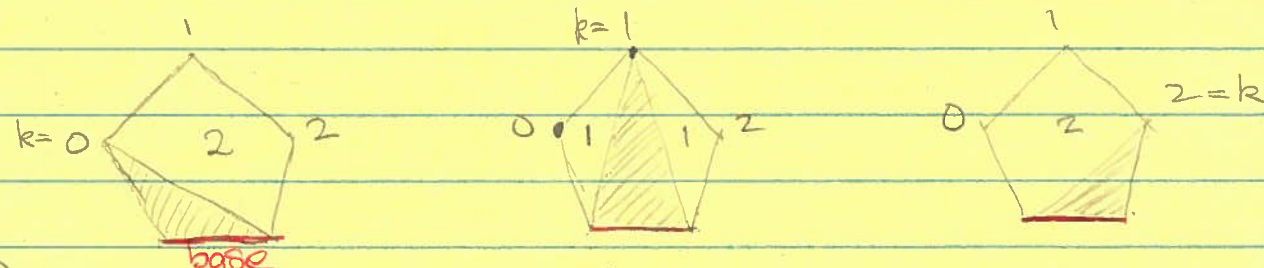
We will see later that the #ways to divide an $(n+2)$-gon is C_n, the n^{th} Catalan number [cf L14]

Prop $$C_n = \sum_{k=0}^{n-1} C_k C_{n-k-1}$$ [Segner recurrence]

Note: the formula above invokes C_0, which we define to be 1 since I don't know what a 2-gon is! [Of course the formula $C_n = \frac{1}{n+1}\binom{2n}{n}$ has no such problem!]

n	0	1	2	3	…
C_n	1	1	2	5	…

Pf case: $n=3$



k (shaded Δ)	0	1	2
#triangulations	$1\cdot 2$	$1\cdot 1$	$2\cdot 1$
	$C_0\cdot C_2$	$C_1\cdot C_1$	$C_2\cdot C_0$

In general, an $(n+2)$-gon has $n+2$ vertices. Omit the two vertices that at either end of the base. That leaves n vertices which we may label $\{0, 1, \ldots, n-1\}$.

For each $k \in \{0, 1, \ldots, n-1\}$ form the shaded triangle.

Count the # triangulations of polygon "to the left" of shaded triangle. This polygon has $k+2$ vertices and therefore C_k triangulations.

Count the # triangulations of polygon "to the right" of shaded triangle. This polygon has $(n+2)-(k+2)+1 = n-k+1$ vertices and therefore C_{n-k-1} triangulations.

Since the triangulations in the "left" and "right" polygons can be formed independently, we have:

$$C_n = \sum_{k=0}^{n-1} C_k C_{n-k-1}.$$

▨

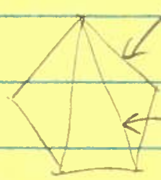
Prop. $(4n+2)\,C_n = (n+2)\,C_{n+1}$. [Simpler recursion] $(*)$

Pf Give a bijection between two sets with $(4n+2)\,C_n$ and $(n+2)\,C_{n+1}$ elements.

The first set is the set of all triangulations of an $(n+2)$-gon in which one edge has been given a direction:

(i) How many triangulations? C_n

(ii) How many edges are in a triangulation?



There are $n+2$ external edges, but how many internal edges are there?

# external	3	4	5	6	7	⋯	n
# internal	0	1	2	3	4	⋯	$n-2-1=n-3$

adjacent external edges

In general an $(n+2)$-gon has $(n+2)-3 = n-1$ internal edges.

Thus:

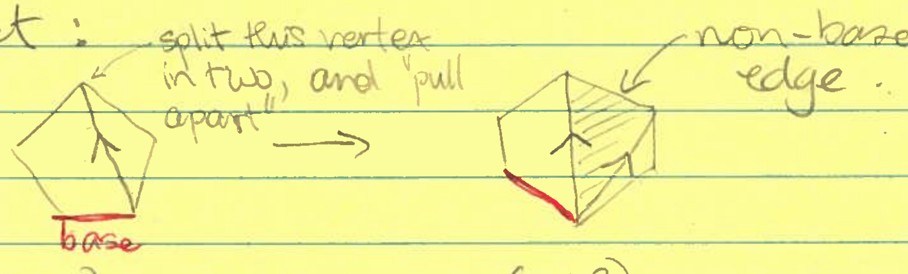
total # edges in triangulation is:

$$(n+2)+(n-1)=2n+1$$

(iii) Finally how many ways to ascribe a direction to a given edge? 2

Multiplying (i), (ii) and (iii) gives LHS of (*).

We now describe an algorithm to transform each element of the above set into an element of a different set:



$(n+2)$-gon $\quad\quad$ $(n+3)$-gon

Clearly, this operation is reversible, so the two sets have the same cardinality.

How many elements in 2nd set?

(i) There are C_{n+1} triangulations

(ii) There are $(n+3)-1=n+2$ ways to

choose a "non-base" side.

Multiplying (i) and (ii) gives RHS of (*).

$\boxtimes$

It is easy to show that our original formula, $C_n = \frac{1}{n+1}\binom{2n}{n}$, solves the recursion (*):

$$(n+2)\,C_{n+1} = (n+2)\frac{1}{(n+1)+1}\cdot\binom{2(n+1)}{n+1}$$

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

$$= \frac{(2n+2)(2n+1)(2n)!}{(n+1)n!\cdot(n+1)n!}$$

$$= \frac{2\cancel{(n+1)}(2n+1)}{\cancel{(n+1)}(n+1)}\binom{2n}{n}$$

$$= (4n+2)\frac{1}{n+1}\binom{2n}{n}$$

$$= (4n+2)\,C_n.$$

Thus the # minimal length paths on an $n \times n$ grid that lie on or below the main diagonal is the same as the

ways to triangulate an $(n+2)$-gon!

In fact there are many other things that Catalan numbers count.

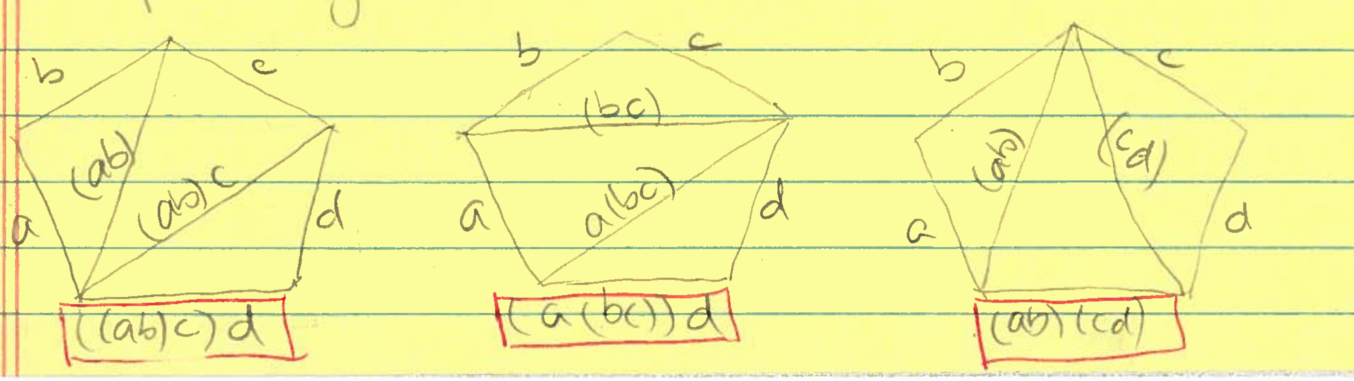
Suppose you want to multiply 4 numbers in a given order:

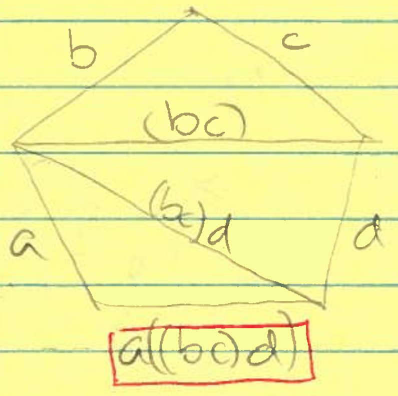
$$a \cdot b \cdot c \cdot d.$$

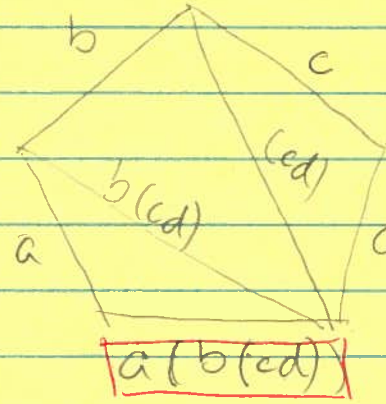
Multiplication is a binary operator, i.e. can only multiply a pair of numbers at a time.

<u>Example</u> Multiply a and b, then multiply the product ab by c giving $(ab)c$, which we then multiply by d, yielding $((ab)c)d$.

In fact, there are 5 ways to parenthesize 3 multiplications, and these are in one-to-one correspondence with triangulations of a pentagon:

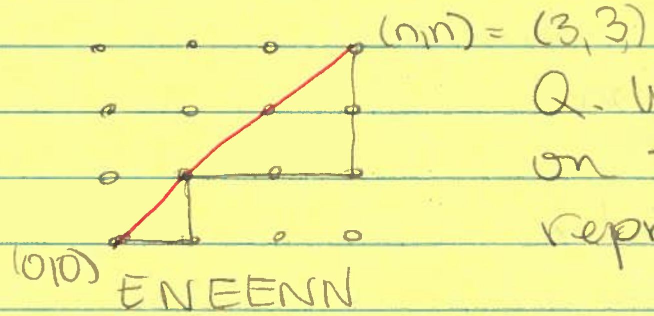






In general, # ways to parenthesize n binary operations among $n+1$ operands is equal to # triangulations of an $(n+2)$-gon, which is C_n.

Recall the minimal length paths that lie on or below the main diagonal:



Q. What is a condition on the string $x_1 x_2 \cdots x_{2n}$ representing such a path?

A. Consider a point (x,y) on the path. Then (x,y) must lie on or below the main diagonal, ie. $x \geq y$.
But $x = $ #E's in the subpath from $(0,0)$ to (x,y) and $y = $ #N's in the subpath [This is so because we forbid W and S moves.]

Thus a condition on a string $x_1 \cdots x_{2n}$ that represents such a path is that at every point in the string, the preceding #E's is never smaller than the #N's.

Such words are called "Dyck words".

Since # Dyck words of length $2n$ and # parenthesizations of n binary operators are both equal to C_n, there must be a bijection between the two sets.

(Non-graded) H/W: describe such a bijection.

A different, but more natural, bijection between Dyck words and parentheses is one where

$$E \leftrightarrow ($$
$$N \leftrightarrow)$$

eg.

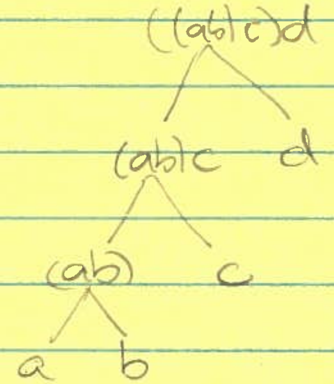
ENEENN $\leftrightarrow$ ()(())

Under this bijection, a Dyck word of length $2n$ corresponds to an expression containing n pairs of correctly matched parentheses. The # of such expressions is therefore C_n.

Trees.

① Successive applications of a binary operator can be represented by a tree:

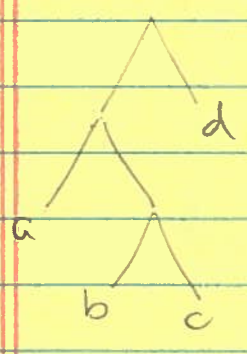
((ab)c)d ⟶



build the tree in this direction ↑

Note: In Computer Science trees like this are used to eg. multiply numbers, parse HTML, parse if-else statements, ...

Similarly, a tree can define a parenthesization:

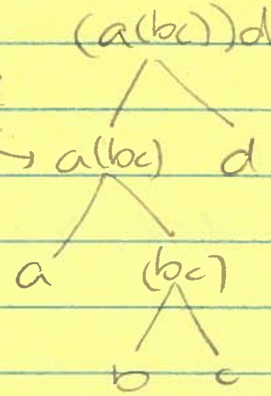


leaf node →

evaluate internal nodes in this direction ↑

⟶

internal node ↘

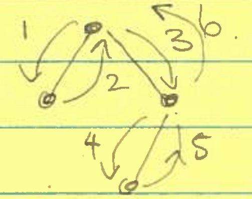


Clearly these trees must be ordered (leaf nodes are a, b, c, d, in that order) and full (internal nodes have exactly two children, reflecting the fact that binary operators have 2 operands).

There are C_n such trees with $(n+1)$ leaves (and therefore n binary operators).

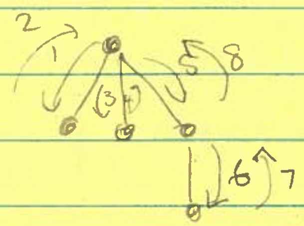
(2) Dyck words can also generate trees:

1 2 3 4 5 6

$\underbrace{ENEENN}_{2n=6}$ $\longrightarrow$  "depth-first" search.

nodes = 4 = n+1

The construction is reversible:

 $\longrightarrow$ 1 2 3 4 5 6 7 8

$\underbrace{ENENEENN}_{2n=8}$

#nodes = 5 = n+1.

Thus C_n counts the # trees with n+1 nodes (internal + leaf).