# Algebraic Data Types

Hype for Types

January 24, 2023

		~		
- H \//	ne '	tor	1.1	nec
- I I V I	De -	IOI -	- I V	DES

Algebraic Data Types

January 24, 2023

イロト イポト イヨト イヨト

• Look at types we already know from a different angle

2

イロト イヨト イヨト

## Outline

- Look at types we already know from a different angle
- Formalize some important new type concepts

э

Image: A matrix

# Outline

- Look at types we already know from a different angle
- Formalize some important new type concepts
- break the universe

э

Image: A matrix

### Introduction to Counting

inc.	tor	1 1 / 1	000
			100
, pe			
		_	

æ

3/36



#### Be prepared to learn some very serious math such as

#### 1 + 2 = 3

	~	_	
<b>n</b> o	tor	1.00	00
 De -	TO I	1 V D	es
		· · / F	

Image: A matrix

# bool and order

Notation

Write  $|\tau|$  to denote the number of elements in type  $\tau^a$ .

<sup>a</sup>this does not work quite well with polymorphism unfortunately.

datatype bool = false | true datatype order = LESS | EQUAL | GREATER What size are they?

# bool and order

Notation

Write  $|\tau|$  to denote the number of elements in type  $\tau^a$ .

<sup>a</sup>this does not work quite well with polymorphism unfortunately.

datatype bool = false | true datatype order = LESS | EQUAL | GREATER What size are they?

> $|\mathbf{bool}| = 2$  $|\mathbf{order}| = 3$

			_	
	-	tor	1.1	DOC
_ I I V	De	101	- I V	Des

3

# bool and order

Notation

Write  $|\tau|$  to denote the number of elements in type  $\tau^a$ .

<sup>a</sup>this does not work quite well with polymorphism unfortunately.

datatype bool = false | true datatype order = LESS | EQUAL | GREATER What size are they?

> $|\mathbf{bool}| = 2$  $|\mathbf{order}| = 3$

Often, we refer to **bool** as 2 and **order** as 3:

- true : 2
- LESS:3

1.1.1	<b>.</b>	+ -	-	 <u> </u>

3

6 / 36

< ロ > < 回 > < 回 > < 回 > < 回 >

Question

What is  $|\tau_1 \times \tau_2|$ ?

		~	_	
	<b>n</b> o	tor	1 1 / 12	000
110	De.	TOT	1 VI.	185

2

<ロト < 四ト < 三ト < 三ト

Question

What is  $|\tau_1 \times \tau_2|$ ?

 $| au_1| imes | au_2|$  - hence, the notation.

	~	_	
<b>n</b> o	tor	1.00	00
 De -	TO I	1 V D	es
		· · / F	

2

イロト イボト イヨト イヨト

#### Question

What is  $|\tau_1 \times \tau_2|$ ?

 $|\tau_1| \times |\tau_2|$  - hence, the notation.

For example,

$$|\mathbf{bool} \times \mathbf{order}| = |\mathbf{bool}| \times |\mathbf{order}|$$
  
= 2 × 3  
= 6

2

7/36

イロト イボト イヨト イヨト

## What do you know!

Theorem: Commutativity of Products

For all  $\tau_1, \tau_2$ :

 $\tau_1 \times \tau_2 \simeq \tau_2 \times \tau_1$ 

Theorem: Associativity of Products For all  $\tau_1, \tau_2, \tau_3$ :  $\tau_1 \times (\tau_2 \times \tau_3) \simeq (\tau_1 \times \tau_2) \times \tau_3$ 

э

イロト 不得 トイヨト イヨト

## What do you know!

Theorem: Commutativity of Products

For all  $\tau_1, \tau_2$ :

 $\tau_1 \times \tau_2 \simeq \tau_2 \times \tau_1$ 

Theorem: Associativity of Products For all  $\tau_1, \tau_2, \tau_3$ :  $\tau_1 \times (\tau_2 \times \tau_3) \simeq (\tau_1 \times \tau_2) \times \tau_3$ 

Question

How do we know?

Hype for Types	Algebraic Data Types	January 24, 2023	8 / 36
	4	민 돈 옷 댄 돈 옷 문 돈 옷 문 돈	≣ ≁) Q (*

# Proving Type Isomorphisms

To prove that  $\tau \simeq \tau'$ , we need a *bijection* between  $\tau$  and  $\tau'$ .

	~	_	
<b>n</b> o	tor	1.00	00
 De -	TO I	1 V D	es
		· · / F	

э

Image: A matrix

# Proving Type Isomorphisms

To prove that  $\tau \simeq \tau'$ , we need a *bijection* between  $\tau$  and  $\tau'$ .

We write two (total) functions,  $f : \tau \to \tau'$  and  $f' : \tau' \to \tau$ , such that f and f' are *inverses*.

f' (f x)  $\cong$  x f (f' x)  $\cong$  x

	<b>n</b> o	tor	Lymon.
110	De.	TOF	TVDES

# Associativity of Products: Proved!

Let's prove associativity of products:

$$au_1 imes ( au_2 imes au_3) \simeq ( au_1 imes au_2) imes au_3$$

		<i>c</i>	_	
- HN	ne	tor	1 1/1	100
	De			163

Image: A matrix

### Associativity of Products: Proved!

Let's prove associativity of products:

$$au_1 imes ( au_2 imes au_3) \simeq ( au_1 imes au_2) imes au_3$$

Need to write:

$$f: \tau_1 \times (\tau_2 \times \tau_3) \to (\tau_1 \times \tau_2) \times \tau_3$$
  
$$f': (\tau_1 \times \tau_2) \times \tau_3 \to \tau_1 \times (\tau_2 \times \tau_3)$$

3 × < 3 ×

Image: A matrix

### Associativity of Products: Proved!

Let's prove associativity of products:

$$au_1 imes ( au_2 imes au_3) \simeq ( au_1 imes au_2) imes au_3$$

Need to write:

$$f: \tau_1 \times (\tau_2 \times \tau_3) \to (\tau_1 \times \tau_2) \times \tau_3$$
  
$$f': (\tau_1 \times \tau_2) \times \tau_3 \to \tau_1 \times (\tau_2 \times \tau_3)$$

Nice!

$$f = fn (a, (b, c)) \Rightarrow ((a, b), c)$$
  
 $f' = fn ((a, b), c) \Rightarrow (a, (b, c))$ 

Hype for Types

Image: A matrix

# Multiplicative Identity?

Follow-Up

Is there an identity element, "1"?

 $\begin{aligned} \tau \times \mathbf{1} &= \tau \\ \mathbf{1} \times \tau &= \tau \end{aligned}$ 

		<i>c</i>	_	
- HM	ne	tor	1.1/1	200
	De			763

э

イロト イボト イヨト イヨト

# Multiplicative Identity?

Follow-Up

Is there an identity element, "1"?

 $\begin{aligned} \tau \times \mathbf{1} &= \tau \\ \mathbf{1} \times \tau &= \tau \end{aligned}$ 

Yes - unit!

		<i>c</i>	_	
- HM	ne	tor	1.17	nec
	De			005

3

イロト イボト イヨト イヨト

# Multiplicative Identity?

Follow-Up

Is there an identity element, "1"?

 $\begin{aligned} \tau \times \mathbf{1} &= \tau \\ \mathbf{1} \times \tau &= \tau \end{aligned}$ 

Yes - unit!



	-		
100	tor	1100	00
 			-
p C			
		_	

æ

<ロ> <四> <ヨ> <ヨ>

### Increment

Question

Is there such thing as  $\tau + 1$ ?

		<i>c</i>	_	
- HN	ne	tor	1 1/1	100
	De			163

æ

A D N A B N A B N A B N

### Increment

#### Question

Is there such thing as  $\tau + 1$ ?

#### Answer

Yes!  $\tau$  option.

		<i>c</i>	_	
- HM	ne	tor	1.1/1	200
	De			763

2

イロト イポト イヨト イヨト

### Increment

#### Question

Is there such thing as  $\tau + 1$ ?

#### Answer

Yes!  $\tau$  option.

SOME x	( au choices)
NONE	(1 choice)

		_	
ne.	tor	LVD66	
De.		1 1 0 0 6 3	

イロト イポト イヨト イヨト

2

datatype ('a,'b) either = Left of 'a | Right of 'b<sup>1</sup>

<sup>1</sup>In the Standard ML Basis, (almost) the Either structure!  $\langle \mathcal{A} \rangle \land \langle \mathbb{R} \rangle \land \langle \mathbb{R} \rangle \land \langle \mathbb{R} \rangle$ 

datatype ('a,'b) either = Left of 'a | Right of 'b<sup>1</sup>

$$\frac{\Gamma \vdash e : \tau_1}{\Gamma \vdash \text{Left } e : \tau_1 + \tau_2} \text{ (LEFT)} \qquad \frac{\Gamma \vdash e : \tau_2}{\Gamma \vdash \text{Right } e : \tau_1 + \tau_2} \text{ (RIGHT)}$$

<sup>1</sup>In the Standard ML Basis, (almost) the Either structure!

datatype ('a,'b) either = Left of 'a | Right of 'b<sup>1</sup>

$$\frac{\Gamma \vdash e : \tau_1}{\Gamma \vdash \text{Left } e : \tau_1 + \tau_2} \text{ (LEFT)} \qquad \frac{\Gamma \vdash e : \tau_2}{\Gamma \vdash \text{Right } e : \tau_1 + \tau_2} \text{ (RIGHT)}$$

$$\frac{\Gamma \vdash e : \tau_1 + \tau_2 \qquad \Gamma, x_1 : \tau_1 \vdash e_1 : \tau \qquad \Gamma, x_2 : \tau_2 \vdash e_2 : \tau}{\Gamma \vdash \textbf{case } e \text{ of } x_1 \Rightarrow e_1 \mid x_2 \Rightarrow e_2 : \tau}$$
(CASE)

<sup>1</sup>In the Standard ML Basis, (almost) the Either structure!  $\langle \square \rangle \land \exists \rangle \land \exists \rangle \land \exists \rangle$ 

	~	
VDO 1	0.5	1000
ivije –		vue
		, – –

datatype ('a,'b) either = Left of 'a | Right of 'b<sup>1</sup>

$$\frac{\Gamma \vdash e : \tau_1}{\Gamma \vdash \text{Left } e : \tau_1 + \tau_2} \text{ (LEFT)} \qquad \frac{\Gamma \vdash e : \tau_2}{\Gamma \vdash \text{Right } e : \tau_1 + \tau_2} \text{ (RIGHT)}$$

$$\frac{\Gamma \vdash e : \tau_1 + \tau_2 \qquad \Gamma, x_1 : \tau_1 \vdash e_1 : \tau \qquad \Gamma, x_2 : \tau_2 \vdash e_2 : \tau}{\Gamma \vdash \mathbf{case} \ e \ \mathbf{of} \ x_1 \Rightarrow e_1 \mid x_2 \Rightarrow e_2 : \tau}$$
(CASE)

And of course...

For all  $\tau_1, \tau_2$ :

$$|\tau_1 + \tau_2| = |\tau_1| + |\tau_2|$$

<sup>1</sup>In the Standard ML Basis, (almost) the Either structure!

Hype for Types	Algebraic Data Types	January 24, 2023	14 / 36

```
datatype ('a, 'b) either = Left of 'a | Right of 'b
```

Notice:

type 'a option = ('a,unit) either

We can represent  $\tau$  option as  $\tau$  + unit.

▲□▶ ▲□▶ ▲□▶ ▲□▶ □ ののの

#### Claim

For all types A, B, C:

$$(A \times B) + (A \times C) \simeq A \times (B + C)$$

		<i>c</i>	_	
- HM	ne	tor	11/1	000
	De			,65

3

イロン 不聞 とくほとう ほとう

#### Claim

For all types A, B, C:

#### $(A \times B) + (A \times C) \simeq A \times (B + C)$

3

イロト イポト イヨト イヨト

#### Claim

For all types A, B, C:

$$(A \times B) + (A \times C) \simeq A \times (B + C)$$

 $f = \text{fn Left (a,b)} \Rightarrow (a, \text{Left b)} | \text{Right (a,c)} \Rightarrow (a, \text{Right c})$  $f' = \text{fn (a, \text{Left b)}} \Rightarrow \text{Left (a,b)} | (a, \text{Right c}) \Rightarrow \text{Right (a,c)}$ 

16/36

▲□▶ ▲□▶ ▲□▶ ▲□▶ □ ののの

#### Claim

For all types A, B, C:

$$(A \times B) + (A \times C) \simeq A \times (B + C)$$

 $f = \text{fn Left (a,b)} \Rightarrow (a, \text{Left b)} | \text{Right (a,c)} \Rightarrow (a, \text{Right c})$  $f' = \text{fn (a, \text{Left b)}} \Rightarrow \text{Left (a,b)} | (a, \text{Right c}) \Rightarrow \text{Right (a,c)}$ 

#### Practical Application

Code refactoring principle! If both cases store the same data, factor it out.

Hype for Types

Algebraic Data Types
If we can add, what's 0?

2

・ロト ・四ト ・ヨト ・ヨト

If we can add, what's 0?

We call it **void**, the empty type.<sup>2</sup>

#### $^{2}\mbox{Unlike C's void type, which is actually unit.}$

If we can add, what's 0?

We call it **void**, the empty type.<sup>2</sup>

**void** is a type which has no value (terminology is *uninhabited*). How do we construct a type with no value (in SML)?

$$\frac{\Gamma \vdash e : \mathsf{void}}{\Gamma \vdash \mathsf{absurd}(e) : \tau} (\text{ABSURD})$$

<sup>2</sup>Unlike C's void type, which is actually **unit**.

Image: A matrix

If we can add, what's 0?

We call it **void**, the empty type.<sup>2</sup>

**void** is a type which has no value (terminology is *uninhabited*). How do we construct a type with no value (in SML)?

$$\frac{\Gamma \vdash e : \mathsf{void}}{\Gamma \vdash \mathsf{absurd}(e) : \tau} (\text{Absurd})$$

Implementing via SML Hacking

datatype void = Void of void
fun absurd (Void v) = absurd v

Notice: absurd is total!

<sup>2</sup>Unlike C's void type, which is actually **unit**.

		_	
 1000	tor	<b>Tb</b> <sub>2</sub>	nor
vue	101		DES

э

(日)

void\*

### Claim

For all types  $\tau$ :

#### $\tau + {\rm void} \simeq \tau$

		~		
- H \//	ne '	tor	1.1	nec
- I I V I	De -	IOI -	- I V	DES

3

イロト イヨト イヨト イヨト

void\*

#### Claim

For all types  $\tau$ :

#### $\tau + {\rm void} \simeq \tau$

f:('tau,void) either -> 'tau
f':'tau -> ('tau,void) either

HV	ne tor	Vnes	
		1 y p C S	

3

イロト 不得 トイヨト イヨト

void\*

Claim

For all types  $\tau$ :

$$\tau + \operatorname{void} \simeq \tau$$

$$f = \text{fn Left } x \Rightarrow x | \text{Right } v \Rightarrow \text{absurd } v$$
  
 $f' = \text{fn } x \Rightarrow \text{Left } x$   
 $= \text{Left}$ 

3

イロト イヨト イヨト イヨト

## Functions

VDA .	tor	1 VD	00
IVDE			

<ロト <問ト < 目と < 目と

3

How many (total) values are there of type  $A \rightarrow B$ , in terms of |A| and |B|?

э

→ < ∃ →</p>

Image: A matrix and a matrix

How many (total) values are there of type  $A \rightarrow B$ , in terms of |A| and |B|?

• How many choices for output of first object of type A?

Image: A matrix

How many (total) values are there of type  $A \rightarrow B$ , in terms of |A| and |B|?

• How many choices for output of first object of type A? |B|

Image: A matrix

How many (total) values are there of type  $A \rightarrow B$ , in terms of |A| and |B|?

- How many choices for output of first object of type A? |B|
- How many choices for the output of the second?

How many (total) values are there of type  $A \rightarrow B$ , in terms of |A| and |B|?

- How many choices for output of first object of type A? |B|
- How many choices for the output of the second? |B|

How many (total) values are there of type  $A \rightarrow B$ , in terms of |A| and |B|?

- How many choices for output of first object of type A? |B|
- How many choices for the output of the second? |B|
- By using our cherished Multiplication Principle from concepts ...

How many (total) values are there of type  $A \rightarrow B$ , in terms of |A| and |B|?

- How many choices for output of first object of type A? |B|
- How many choices for the output of the second? |B|
- By using our cherished Multiplication Principle from concepts ...

#### Theorem

There are  $|B|^{|A|}$  total functions from type A to type B.

Example: Power of a Power

In math, it's true that:

$$(C^B)^A = C^{A \times B}$$

(DO)	tor	Lymon.
 /De	101	TVDES

< ∃⇒

э

### Example: Power of a Power

In math, it's true that:

$$(C^B)^A = C^{A \times B}$$

In terms of types, that would mean:

$$A 
ightarrow (B 
ightarrow C) \simeq A imes B 
ightarrow C$$

		<i>c</i>	_	
- HM	ne	tor	1.1/1	200
	De			763

э

### Example: Power of a Power

In math, it's true that:

$$(C^B)^A = C^{A \times B}$$

In terms of types, that would mean:

$$A \rightarrow (B \rightarrow C) \simeq A \times B \rightarrow C$$

Yes!

э

21/36

# Recursive Types

1.1.1	<b>.</b>	+ -		<u> </u>	
			•••		
				$\sim$	

3

22 / 36

<ロト <問ト < 目と < 目と

datatype 'a list = Nil | Cons of 'a \* 'a list

datatype 'a list = Nil | Cons of 'a \* 'a list

datatype 'a list = Left of unit | Right of 'a \* 'a list

		~	_	
	<b>n</b> o	tor	1 1 / 12	000
110	De.	TOT	1 VI.	185

datatype 'a list = Nil | Cons of 'a \* 'a list

datatype 'a list = Left of unit | Right of 'a \* 'a list

type 'a list = (unit, 'a \* 'a list) either

		<i>c</i>	_	
H N	ne	tor	1.17	nec
	De			Des

<□> <同> <同> < 回> < 回> < 回> < 回> < 回> < □> < □> ○ Q ()

datatype 'a list = Nil | Cons of 'a \* 'a list

datatype 'a list = Left of unit | Right of 'a \* 'a list

type 'a list = (unit, 'a \* 'a list) either

 $L(\alpha) \simeq \text{unit} + \alpha \times L(\alpha)$ 

datatype 'a list = Nil | Cons of 'a \* 'a list

datatype 'a list = Left of unit | Right of 'a \* 'a list

type 'a list = (unit, 'a \* 'a list) either

 $L(\alpha) \simeq \text{unit} + \alpha \times L(\alpha)$ 

$$L(\alpha) = 1 + \alpha \times L(\alpha)$$
  
= 1 + \alpha \times (1 + \alpha \times L(\alpha))  
= 1 + \alpha + \alpha \times L(\alpha)  
= 1 + \alpha + \alpha^2 + \alpha^3 + \dots

How many natural numbers are there?

э

・ロト ・四ト ・ヨト ・ヨト

How many natural numbers are there?

datatype nat = Zero | Succ of nat

Hype for Types	Algebraic Data Types	Jai

How many natural numbers are there?

datatype nat = Zero | Succ of nat

nat = unit + nat

		<i>c</i>		
- HV	ne	tor	LVD66	
	De		1 1 0 0 6 3	

э

イロト イボト イヨト イヨト

How many natural numbers are there?

datatype nat = Zero | Succ of nat

nat = unit + nat

 $\mathsf{nat} = 1 + 1 + 1 + \dots = \infty$ 

3

イロト 不得 トイヨト イヨト

How many natural numbers are there?

datatype nat = Zero | Succ of nat

nat = unit + nat

 $\mathsf{nat} = 1 + 1 + 1 + \dots = \infty$ 

Therefore, we would expect:

 $\infty = 1 + \infty$ 

#### nat $\simeq$ nat option

(日)

3

How many natural numbers are there?

datatype nat = Zero | Succ of nat

nat = unit + nat

 $nat = 1 + 1 + 1 + \dots = \infty$ 

Therefore, we would expect:

 $\infty = 1 + \infty$ 

#### nat $\simeq$ nat option

f = fn Zero => NONE | Succ n => SOME n f' = fn NONE => Zero | SOME n => Succ n

Image: A matrix

3

### **Binary Trees**

```
datatype 'a tree
 = Empty
 | Node of 'a tree * 'a * 'a tree
```

## **Binary Trees**

#### datatype 'a tree = Empty | Node of 'a tree \* 'a \* 'a tree

$$T(\alpha) \simeq \text{unit} + T(\alpha) \times \alpha \times T(\alpha)$$
  
 $\simeq \text{unit} + \alpha \times T(\alpha)^2$ 

Hype for Types

Algebraic Data Types

January 24, 2023

25 / 36

## **Binary Shrubs**

```
datatype 'a shrub
= Leaf of 'a
| Node of 'a shrub * 'a shrub
```

### **Binary Shrubs**

```
datatype 'a shrub
= Leaf of 'a
| Node of 'a shrub * 'a shrub
```

$$egin{aligned} \mathcal{S}(lpha) &\simeq lpha + \mathcal{S}(lpha) imes \mathcal{S}(lpha) \ &\simeq lpha + \mathcal{S}(lpha)^2 \end{aligned}$$

# Counting

How many binary shrubs are there?

3

イロト イヨト イヨト イヨト

# Counting

How many binary shrubs are there?

$$S(\alpha) = \alpha + S(\alpha)^2$$

3

イロト イヨト イヨト イヨト
# Counting

How many binary shrubs are there?

$$S(\alpha) = \alpha + S(\alpha)^2$$
$$0 = S(\alpha)^2 - S(\alpha) + \alpha$$

2

イロト イヨト イヨト

# Counting

How many binary shrubs are there?

$$S(lpha) = lpha + S(lpha)^2$$
  
 $0 = S(lpha)^2 - S(lpha) + lpha$   
 $S(lpha) = rac{1 - \sqrt{1 - 4lpha}}{2}$  (quadratic formula)

			_	
Hv	ne	tor	1 3 / 7	100
	De			, – –

Image: A matched block

# Counting

How many binary shrubs are there?

$$S(\alpha) = \alpha + S(\alpha)^{2}$$

$$0 = S(\alpha)^{2} - S(\alpha) + \alpha$$

$$S(\alpha) = \frac{1 - \sqrt{1 - 4\alpha}}{2} \qquad (\text{quadratic formula})$$

$$S(\alpha) = \alpha^{1} + \alpha^{2} + 2\alpha^{3} + 5\alpha^{4} + \ldots + \frac{1}{n} \binom{2n - 2}{n - 1} \alpha^{n} + \ldots$$
(Taylor series)

3

イロト イヨト イヨト イヨト

$$S(\alpha) = \alpha^1 + \alpha^2 + 2\alpha^3 + 5\alpha^4 + \ldots + \frac{1}{n} {\binom{2n-2}{n-1}} \alpha^n + \ldots$$

3

・ロト ・四ト ・ヨト ・ヨト

$$S(\alpha) = \alpha^1 + \alpha^2 + 2\alpha^3 + 5\alpha^4 + \ldots + \frac{1}{n} {\binom{2n-2}{n-1}} \alpha^n + \ldots$$

• Each leaf has  $\alpha$  choices for its value

< □ > < 同 > < 回 > < 回 > < 回 >

э

$$S(\alpha) = \alpha^1 + \alpha^2 + 2\alpha^3 + 5\alpha^4 + \ldots + \frac{1}{n} \binom{2n-2}{n-1} \alpha^n + \ldots$$

- Each leaf has  $\alpha$  choices for its value
- Any 1 leaf shrub form would contribute  $\alpha^1$  to the count

	c .	_
VDA	tor	LVD66
IV DE		1 1 1 1 2 3

$$S(\alpha) = \alpha^1 + \alpha^2 + 2\alpha^3 + 5\alpha^4 + \ldots + \frac{1}{n} \binom{2n-2}{n-1} \alpha^n + \ldots$$

- Each leaf has  $\alpha$  choices for its value
- Any 1 leaf shrub form would contribute  $\alpha^1$  to the count
- Any 4 leaf shrub form would contribute  $\alpha^4$  to the count

$$S(\alpha) = \alpha^1 + \alpha^2 + 2\alpha^3 + 5\alpha^4 + \ldots + \frac{1}{n} \binom{2n-2}{n-1} \alpha^n + \ldots$$

- Each leaf has  $\alpha$  choices for its value
- Any 1 leaf shrub form would contribute  $\alpha^1$  to the count
- Any 4 leaf shrub form would contribute  $\alpha^4$  to the count

#### Revelation

```
\frac{1}{n}\binom{2n-2}{n-1} is the number of 'a shrubs of n nodes!
```

$$S(\alpha) = \alpha^1 + \alpha^2 + 2\alpha^3 + 5\alpha^4 + \ldots + \frac{1}{n} \binom{2n-2}{n-1} \alpha^n + \ldots$$

- Each leaf has  $\alpha$  choices for its value
- Any 1 leaf shrub form would contribute  $\alpha^1$  to the count
- Any 4 leaf shrub form would contribute  $\alpha^4$  to the count

#### Revelation

```
\frac{1}{n}\binom{2n-2}{n-1} is the number of 'a shrubs of n nodes!
```

#### • This sequence is called the Catalan numbers

28 / 36

$$S(\alpha) = \alpha^1 + \alpha^2 + 2\alpha^3 + 5\alpha^4 + \ldots + \frac{1}{n} \binom{2n-2}{n-1} \alpha^n + \ldots$$

- Each leaf has  $\alpha$  choices for its value
- Any 1 leaf shrub form would contribute  $\alpha^1$  to the count
- Any 4 leaf shrub form would contribute  $\alpha^4$  to the count

#### Revelation

```
\frac{1}{n}\binom{2n-2}{n-1} is the number of 'a shrubs of n nodes!
```

- This sequence is called the Catalan numbers
- This technique is called Generating Functions

#### haha type derivates go brrr

	_	_	
 <b>n</b> o 1	O.F.	1.00	200
			100
 PC			

2

29 / 36

イロト イヨト イヨト

## Taking Things Too Far

Question

What is  $\frac{d}{d\alpha}\tau(\alpha)$ ?

	~	_	
<b>n</b> o	tor	1.00	00
 De -	TO I	1 V D	es
		· · / F	

2

イロト イポト イヨト イヨト

# Taking Things Too Far

#### Question

What is  $\frac{d}{d\alpha}\tau(\alpha)$ ?

#### Smart Idea

Dismiss the idea outright - this is madness!

		<i>c</i>	_	
HV	ne	tor	1.17	nec
	De			Des

э

(日)

# Taking Things Too Far

#### Question

What is  $\frac{d}{d\alpha}\tau(\alpha)$ ?

#### Smart Idea

Dismiss the idea outright - this is madness!

#### Our Plan

>:)

Hype for Types	Algebraic Data Types	January 24, 2023	30 / 36
		(日本・御をする)、日本	E 990



$$\frac{d}{d\alpha}\alpha^{3} = \left(\frac{d}{d\alpha}\alpha \times \alpha \times \alpha\right) + \left(\alpha \times \frac{d}{d\alpha}\alpha \times \alpha\right) + \left(\alpha \times \alpha \times \frac{d}{d\alpha}\alpha\right)$$

Ξ.

◆□ ▶ ◆圖 ▶ ◆ 臣 ▶ ◆ 臣 ▶ ○



$$\frac{d}{d\alpha}\alpha^{3} = \left(\frac{d}{d\alpha}\alpha \times \alpha \times \alpha\right) + \left(\alpha \times \frac{d}{d\alpha}\alpha \times \alpha\right) + \left(\alpha \times \alpha \times \frac{d}{d\alpha}\alpha\right)$$
$$\frac{d}{d\alpha}\alpha^{3} = 3\alpha^{2}$$

◆□ ▶ ◆圖 ▶ ◆ 臣 ▶ ◆ 臣 ▶ ○

Ξ.



$$\frac{d}{d\alpha}\alpha^{3} = \left(\frac{d}{d\alpha}\alpha \times \alpha \times \alpha\right) + \left(\alpha \times \frac{d}{d\alpha}\alpha \times \alpha\right) + \left(\alpha \times \alpha \times \frac{d}{d\alpha}\alpha\right)$$
$$\frac{d}{d\alpha}\alpha^{3} = 3\alpha^{2}$$
$$\alpha \times \alpha \times \alpha \quad \mapsto \qquad 3 \times (\alpha \times \alpha)$$

・ロト ・四ト ・ヨト ・ヨト 三日



$$\frac{d}{d\alpha}\alpha^{3} = \left(\frac{d}{d\alpha}\alpha \times \alpha \times \alpha\right) + \left(\alpha \times \frac{d}{d\alpha}\alpha \times \alpha\right) + \left(\alpha \times \alpha \times \frac{d}{d\alpha}\alpha\right)$$
$$\frac{d}{d\alpha}\alpha^{3} = 3\alpha^{2}$$
$$\alpha \times \alpha \times \alpha \qquad \mapsto \qquad 3 \times (\alpha \times \alpha)$$

Differentiating a power "eats" a tuple slot, and tells you which element was removed.

Hype for Types	Algebraic Data Types	January 24, 2023	31 / 36
	<		≣ • <b>ગ</b> ∢ભ

## Differentiating a List

Recall that:

$$a + ar + ar^2 + ar^3 + \dots = \frac{a}{1-r}$$

<sup>3</sup>What the hype is a negative type?

э

# Differentiating a List

Recall that:

We have:<sup>3</sup>

$$a + ar + ar^{2} + ar^{3} + \dots = \frac{a}{1 - r}$$
$$L(\alpha) = 1 + \alpha + \alpha^{2} + \dots \stackrel{?}{=} \frac{1}{1 - \alpha}$$

<sup>3</sup>What the hype is a negative type?

э

イロト イポト イヨト イヨト

# Differentiating a List

Recall that:

We have:<sup>3</sup>

$$a + ar + ar^{2} + ar^{3} + \dots = \frac{a}{1 - r}$$
$$L(\alpha) = 1 + \alpha + \alpha^{2} + \dots \stackrel{?}{=} \frac{1}{1 - \alpha}$$

$$\frac{d}{d\alpha}L(\alpha) = \frac{d}{d\alpha}\frac{1}{1-\alpha}$$
$$= \frac{1}{(1-\alpha)^2}$$
$$= \left(\frac{1}{1-\alpha}\right)^2$$
$$= L(\alpha)^2$$

<sup>3</sup>What the hype is a negative type?

Hype for Types

Image: A matrix

э

# Tree for Two, and Two for Tree

We said:

$$T(\alpha) = 1 + \alpha T(\alpha)^2$$

Here we go again...

		_	
	mo.	tor	Lymon.
110	De.	101	TVDES

2

・ロト ・四ト ・ヨト ・ヨト

# Tree for Two, and Two for Tree

We said:

$$T(\alpha) = 1 + \alpha T(\alpha)^2$$

Here we go again...

$$\frac{d}{d\alpha}T(\alpha) = \frac{d}{d\alpha}1 + \frac{d}{d\alpha}\alpha T(\alpha)^2$$
$$= \alpha \times \frac{d}{d\alpha}T(\alpha)^2 + \frac{d}{d\alpha}\alpha \times T(\alpha)^2$$
$$= 2\alpha T(\alpha) \times \frac{d}{d\alpha}T(\alpha) + T(\alpha)^2$$
$$\frac{d}{d\alpha}T(\alpha) = T(\alpha)^2 \left(\frac{1}{1 - 2\alpha T(\alpha)}\right)$$
$$= T(\alpha)^2 L(2\alpha T(\alpha))$$

・ロト ・四ト ・ヨト ・ヨト

$$\frac{d}{d\alpha}\alpha^3 = 3\alpha^2$$
$$\frac{d}{d\alpha}L(\alpha) = L(\alpha)^2$$
$$\frac{d}{d\alpha}T(\alpha) = T(\alpha)^2L(2\alpha T(\alpha))$$

3

$$\frac{d}{d\alpha}\alpha^{3} = 3\alpha^{2}$$
$$\frac{d}{d\alpha}L(\alpha) = L(\alpha)^{2}$$
$$\frac{d}{d\alpha}T(\alpha) = T(\alpha)^{2}L(2\alpha T(\alpha))$$

#### Theorem

The Derivative of a Regular Type is its Type of One-Hole Contexts.<sup>a</sup>

<sup>a</sup>http://strictlypositive.org/diff.pdf

			_	
- HN	ne.	tor	1.17	nec
111	ue.	101	- I V	Des

3

$$\frac{d}{d\alpha}\alpha^{3} = 3\alpha^{2}$$
 "punctured" tuple  
$$\frac{d}{d\alpha}L(\alpha) = L(\alpha)^{2}$$
$$\frac{d}{d\alpha}T(\alpha) = T(\alpha)^{2}L(2\alpha T(\alpha))$$

#### Theorem

The Derivative of a Regular Type is its Type of One-Hole Contexts.<sup>a</sup>

<sup>a</sup>http://strictlypositive.org/diff.pdf

		-
UDA	tor	LVD66
 v De	101	I V DES

3

34 / 36

(本語) ト (本語) ト (本語) ト

$$\frac{d}{d\alpha}\alpha^{3} = 3\alpha^{2}$$
 "punctured" tuple  
$$\frac{d}{d\alpha}L(\alpha) = L(\alpha)^{2}$$
 list zipper  
$$\frac{d}{d\alpha}T(\alpha) = T(\alpha)^{2}L(2\alpha T(\alpha))$$

#### Theorem

The Derivative of a Regular Type is its Type of One-Hole Contexts.<sup>a</sup>

<sup>a</sup>http://strictlypositive.org/diff.pdf

			_	
- Hvz	ne	tor	1 V/P	100
	De			, – –

3

34 / 36

$$\frac{d}{d\alpha}\alpha^{3} = 3\alpha^{2}$$
 "punctured" tuple  
$$\frac{d}{d\alpha}L(\alpha) = L(\alpha)^{2}$$
 list zipper  
$$\frac{d}{d\alpha}T(\alpha) = T(\alpha)^{2}L(2\alpha T(\alpha))$$
 tree zipper

#### Theorem

The Derivative of a Regular Type is its Type of One-Hole Contexts.<sup>a</sup>

<sup>a</sup>http://strictlypositive.org/diff.pdf

			_	
L.	mo.	tor	1.0	000
	ue.	101	- I V I	Des

3

34 / 36

	-		
100	tor	1100	$\sim \sim$
 			-
, p.e.			
		_	

3

イロト イヨト イヨト イヨト

• Figured out the sizes of various types

<sup>4</sup>More on that later...

Hype for Types

∃ → January 24, 2023

36 / 36

э

- Figured out the sizes of various types
- Generalized our type theory to include sum types (and void)

<sup>4</sup>More on that later...

- Figured out the sizes of various types
- Generalized our type theory to include *sum types* (and **void**)
- Considered *recursive types*<sup>4</sup>

<sup>4</sup>More on that later...

- Figured out the sizes of various types
- Generalized our type theory to include *sum types* (and **void**)
- Considered *recursive types*<sup>4</sup>
- Used type equations and generating functions to count objects

- Figured out the sizes of various types
- Generalized our type theory to include *sum types* (and **void**)
- Considered *recursive types*<sup>4</sup>
- Used type equations and generating functions to count objects
- Invented a type-level hole punch