Objectif d'Alain : formaliser $\{0, 1, \dots, n-1\}$.

ssreflect / mathcomponent.

$\Gamma \vdash a : A$ est un jugement de typage : a est de type A dans le contexte Γ

$\Gamma \vdash a \equiv b : A$ est un jugement d'égalité : a est égal à b dans le type A, ds Γ.

On dit que a habite dans A.
a est un habitant de A.

Un contexte Γ est une suite finie de jugements de typage : $x_1 : A_1, \dots, x_n : A_n$.

(ici chaque A_i peut être paramétrée par $x_1, \dots, x_{i-1}$)

Γ contexte, « Γ ctx » exprime que Γ est bien formé.

Définir un type, c'est poser 5 sortes de règles : } typage } égalité.

règle de 1- bonne formation : FORM : c'est de la syntaxe ex : « Π-FORM ».

règle de 2- INTROduction. (1 par constructeur d'objet : mettre des habitants ds le type)

règle d' 3- ELIMination : c'est pour clore un type : "pas d'autre élément que ceux construits"
la récurrence fait partie de cette règle.

Un univers : n'est habité que par des types : il n'y a pas d'objet.

Il y a une suite croissante d'univers et U_i habite dans U

$A : U$ exprime que A est un type.

Famille de types C : $C : A \to U$. ici A est un type : $A : U' \vdash C : A \to U$.
→ autant d'univers que d'habitants de A.

Principe d'induction : ind... définit quand une fonction ~~[illegible]~~ de support ... est déterminée

ex : ind_1 : exprime qd une fonction $f : 1 \to C$ est déterminée.
→ ici, qd on connait la valeur de f en $*$.

Présentation dans la partie 1 : définition du produit cartésien.

$f : A \times B \to C$: on passe par la curryfiée de f qui est $g : A \to (B \to C)$
(a, b) et on pose $f((a,b)) = g(a)(b)$.

Type de cette f° dépendante ca d $A \times B$

Exemple : fonction f_1 telle que $f_1(n)$ est le plus ~~grand~~ petit nombre premier ~~$\leq n$~~ $> n$.

$$f_1 : \mathrm{nat} \to \mathrm{nat} \quad (\text{en maths on écrirait } \mathbb{N} \to \mathbb{N}).$$

$$f_1 : (n : \mathrm{nat}) \longrightarrow \underbrace{]n, +\infty[}_{\text{type (à prouver !)}}$$

↑ ici le type dépend du paramètre !
on dit que la f^{ion} est dépendante.

Exemple : décrire toutes les matrices identité. Fonction f_2 telle que $f_2(n)$ est la matrice identité de taille $n \times n$ dans $\mathbb{C}$.

$$f_2 : \mathrm{nat} \longrightarrow \bigcup_{n \geq 0} \mathbb{C}^{n \times n}$$

s'allège en $f_2 : (n : \mathrm{nat}) \longrightarrow \mathbb{C}^{n \times n}$.

Le type des $\boxed{f^{\text{ions}} \text{ dépdtes}}$: ex : $\prod (n : \mathrm{nat})\ \mathbb{C}^{n \times n}$.

"type fonctionnel" $\quad \prod (x : A)\ B(x)$ et ici B est une fonction : $B : A \to \mathcal{U}$.
ici B est une famille de type.

$\prod$ est un objet central de la théorie

Retour au produit cartésien : ~~$f : \prod_{(A \times B)} C(x)$~~ $\quad f : \prod_{(x : A \times B)} C(p_1(x))(p_2(x))$
f^{ion} dépendante à support dans $A \times B$. où $p_1 : A \times B \to A$ projection, $p_2 : A \times B \to B$

Mais il faut préciser le type de $C : A \times B \to \mathcal{U}$, voire $C : A \to (B \to \mathcal{U})$

$$g : \prod_{(a : A)} \Big(\prod_{(b : B)} C((a, b)) \Big)$$

Introduction : $\dfrac{\Gamma \vdash a : A \qquad \Gamma \vdash b : B}{\Gamma \vdash (a, b) : A \times B.}$

Formation : $\dfrac{\Gamma \vdash A : \mathcal{U} \qquad \Gamma \vdash B : \mathcal{U}}{\Gamma \vdash A \times B : \mathcal{U}.}$

~~$\mathrm{ind}_{A \times B} : \prod_{C : A \times B \to \mathcal{U}} ($~~
~~$\mathrm{ind}_{A \times B}(C)(g)(x) : C(x)$.~~
~~$\underbrace{\prod_{x : A \times B} C(x)}_{\text{type}}$.~~
~~$\prod_{(a : A)} \underbrace{\prod_{(b : B)} C((a, b))}_{\text{type}} \longrightarrow \prod_{x : A \times B} C(x)$.~~

$$\mathrm{ind}_{A \times B} : \prod_{C : A \times B \to \mathcal{U}} \Bigg(\Big(\underbrace{\prod_{(a : A)} \prod_{(b : B)} C((a, b))}_{\text{type de } g} \Big) \longrightarrow \underbrace{\prod_{(x : A \times B)} C(x)}_{\text{type résultat}} \Bigg)$$

$$\mathrm{ind}_{A \times B}(C)(g)(x) : C(x)$$

cf A-1-3.

Le type vide : A-2.7.

$f : 0 \to C$: pas de condition pour construire une fonction

| ind_0 exprime qu'il n'y a besoin de rien pour définir 0 :
| 0-ELIM c'est un genre de "ex falso quod libet".

$C : 0 \to \mathcal{U}$.

$\mathrm{ind}_0 : \prod_{C : 0 \to \mathcal{U}} \prod_{(x:0)} C(x)$
pour toute famille de type de 0 dans $\mathcal{U}$, besoin de rien.

Donc ind_0 prend deux arguments : $\mathrm{ind}_0(\underbrace{C, x}_{\text{entrée}}) : \underbrace{C(x)}_{\text{sortie}}$.

Considérons $C : \prod_{(y:0)} \mathcal{U}$:
$\mathrm{ind}_0 : \left(\prod_{\prod(y:0)\mathcal{U}} \left(\prod_{x:0} C(x)\right)\right)$.
$\mathrm{ind}_0(y.C, x) : C[x/y]$

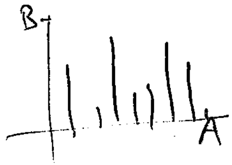
$C : \prod_{(x:0)} \mathcal{U}$
$\mathrm{ind}_0 : \prod_{\prod(x:0)\mathcal{U}} \left(\prod_{a:0} C(a)\right)$
$\mathrm{ind}_0(x.C)(a) : C[a/x]$

Chgt de pt de vue : partie 1 : $\prod_{(a:A)} B(a)$
$B : A \to \mathcal{U}$.

Annexe : on appelle "$B(a)$" "D"
$\prod_{(a:A)} D$ avec $D = B(a)$
i.e. $B = \lambda a.D$
Quel est le type de B et D ?

$\frac{a : A \vdash D : \mathcal{U}}{\Gamma \vdash (\lambda a.D) : A \to \mathcal{U}}$

Exercices : 1) le type unité 1
Trouver le principe d'induction !



2) $A \times B$ se généralise en $\sum_{(x:A)} B$

3) Règle de $\prod_{(x:A)} B$.

5 : $A + B$. (C'est la somme disjointe, le coproduit,
Le coproduit d'anneaux : c'est le produit tensoriel.

6) Les naturels.

7) Le type identité $=_A$ ($\equiv_A$)

et cela termine MLTT