6.

$$\frac{\vdash \Gamma \ ctx}{\Gamma \vdash \mathcal{U}_i \ type} \quad (\mathcal{U}_i\text{-Formation})$$

$$\frac{\Gamma \vdash \alpha : \mathcal{U}_i}{\Gamma \vdash El_i(\alpha) \ type} \quad (\mathcal{U}_i\text{-Elimination})$$

($\mathcal{U}_i$-Introduction rules):

$$\frac{\vdash \Gamma \ ctx}{\Gamma \vdash 0 : \mathcal{U}_0}$$

$$\frac{\vdash \Gamma \ ctx}{\Gamma \vdash \bullet : \mathcal{U}_0}$$

$$\frac{\Gamma \vdash \alpha : \mathcal{U}_i \quad \Gamma, x : El_i(\alpha) \vdash \beta : \mathcal{U}_i}{\Gamma \vdash \pi_i(\alpha, \lambda(x : El_i(\alpha)).\beta) : \mathcal{U}_i}$$

$$\frac{\Gamma \vdash \alpha : \mathcal{U}_i \quad \Gamma, x : El_i(\alpha) \vdash \beta : \mathcal{U}_i}{\Gamma \vdash \sigma_i(\alpha, \lambda(x : El_i(\alpha)).\beta) : \mathcal{U}_i}$$

$$\frac{\Gamma \vdash \alpha : \mathcal{U}_i \quad \Gamma \vdash a : El_i(\alpha) \quad \Gamma \vdash b : El_i(\alpha)}{\Gamma \vdash id^i_\alpha(a, b) : \mathcal{U}_i}$$

$$\frac{\vdash \Gamma \ ctx}{\Gamma \vdash \mathcal{U}_i : \mathcal{U}_{i+1}} \ \text{f.a. } i \in \mathbb{N}$$

$$\frac{\Gamma \vdash \alpha : \mathcal{U}_i}{\Gamma \vdash el_i(\alpha) : \mathcal{U}_{i+1}}$$

($\mathcal{U}_i$-Computation rules):

$$\frac{\vdash \Gamma \ ctx}{\Gamma \vdash El_i(0) \equiv \mathbb{0} \ type}$$

$$\frac{\vdash \Gamma \ ctx}{\Gamma \vdash El_i(\bullet) \equiv \mathbb{1} \ type}$$

$$\frac{\Gamma \vdash \alpha : \mathcal{U}_i \quad \Gamma, x : El_i(\alpha) \vdash \beta : \mathcal{U}_i}{\Gamma \vdash El_i(\pi_i(\alpha, \lambda(x : El_i(\alpha)).\beta)) \equiv \prod_{x : El_i(\alpha)} El_i(\beta) \ type}$$

$$\frac{\Gamma \vdash \alpha : \mathcal{U}_i \quad \Gamma, x : El_i(\alpha) \vdash \beta : \mathcal{U}_i}{\Gamma \vdash El_i(\sigma_i(\alpha, \lambda(x : El_i(\alpha)).\beta)) \equiv \sum_{x : El_i(\alpha)} El_i(\beta) \ type}$$

$$\frac{\Gamma \vdash \alpha : \mathcal{U}_i \quad \Gamma \vdash a : El_i(\alpha) \quad \Gamma \vdash b : El_i(\alpha)}{\Gamma \vdash El(id^i_\alpha(a, b)) \equiv (a =_{El_i(\alpha)} b) \ type}$$

$$\frac{\vdash \Gamma \ ctx}{\Gamma \vdash El_{i+1}(\mathcal{U}_i) \equiv \mathcal{U}_i \ type}$$

$$\frac{\Gamma \vdash \alpha : \mathcal{U}_i}{\Gamma \vdash El_{i+1}(el_i(\alpha)) \equiv El_i(\alpha) \ type}$$

$$\frac{\vdash \Gamma\ ctx}{\Gamma \vdash \tau_T : \mathcal{U}_0}\ \text{f.a. } T \in \Pi \qquad\qquad \frac{\vdash \Gamma\ ctx}{\Gamma \vdash El(\tau_T) = T\ type}\ \text{f.a. } T \in \Pi$$

(+ all other additional type formers one may wish to add to the logical rules)

$\hat{\Gamma}$ These rules define a nested sequence of universes "as uniform constructions" in the sense of Palmgren "On universes in type theory".

Variations in presentations are common, e.g. one may or may not add

$$\frac{\Gamma \vdash \alpha : \mathcal{U}_i \quad \Gamma \vdash \beta : \mathcal{U}_i \quad \Gamma \vdash El_i(\alpha) \equiv El_i(\beta)\ type}{\Gamma \vdash \alpha \equiv \beta : \mathcal{U}_i}\ \text{(Universe-Reflection)}$$

This is implicit to the informal presentation of universes à la Russell.

Exercise 11: 1. Let $A \in Pretype$, $\vdash \Gamma\ ctx$. Then $\Gamma \vdash A\ type$ iff there is a preterm α and an integer $i \geq 0$ s.th. $\Gamma \vdash \alpha : \mathcal{U}_i$ and $\Gamma \vdash El(\alpha) \equiv A\ type$.

↳ Typability judgements can be reduced to judgements of type assignment.

2. Suppose $n \geq 0$, $A_i \in Pretype$ for $0 < i \leq n+1$ s.th. $x_1 : A_1, \dots, x_i : A_i \vdash A_{i+1}\ type$ f.a. $0 < i \leq n$.

Let $\Gamma_i := (x_1 : A_1, \dots, x_i : A_i)$. Then

(a) - $\Gamma_n \vdash a : A_{n+1}$ iff $\emptyset \vdash \lambda(x_1 : A_1). \dots \lambda(x_n : A_n).\ a : \prod_{x_1 : A_1} \dots \prod_{x_n : A_n} A_{n+1}$

- $\Gamma_n \vdash a_1 \equiv a_2 : A_{n+1}$ iff $\emptyset \vdash \lambda \dots a_1 \equiv \lambda \dots a_2 : \prod_{x_1 : A_1} \dots \prod_{x_n : A_n} A_{n+1}$

↳ Single judgements of type assignment and term-congruence can be reduced to the empty context.

(6) $\Phi \vdash a_1 : A_1$, $\Phi \vdash a_2 : A_2[a_1/x_1]$, $\Phi \vdash a_{n+1} : A_{n+1}[\vec{a}/\vec{x}]$ iff

$\Phi \vdash (a_1, \dots, a_{n+1}) : \sum_{x_1 : A_1} \cdots \sum_{x_n : A_n} A_{n+1}$. For families of term congruences accordingly.

↳ Finitely many judgements of type assignment and of term-congruences can be reduced to a single one each.

Corollary 12: 1. Judgements of mathematical practice are of the form "$a : A$" (in ctx $\emptyset$).

2. Every type-family $\Gamma, x : A \vdash B$ type yields a function

$\Gamma \vdash \lambda(x:A).\beta : A \to \mathcal{U}_i$ s.th. $\Gamma, x : A \vdash El(\beta) \equiv B$ type.

This is a 1-1 correspondence (up to judgemental equality) under Universe-Reflection.

Furthermore, in the presence of Π and Σ-types, the Elimination rules can be phrased in terms of induction principles: Given a type A (i.e. $\Phi \vdash A$ type) and a family $\beta : A \to \mathcal{U}_i$, let $x : A \vdash B :\equiv El(App(\beta, x))$. We get by λ-abstraction

$$\lambda(-).\mathrm{ind}_{\sum_{x:A} B} : \prod_{c : \sum_{x:A} B \to \mathcal{U}_j} \left(\prod_{x:A} \prod_{y:B} \overbrace{El(App(c, (x,y)))}^{C[(x,y)/z]} \right) \to \prod_{z : \sum_{x:A} B} \overbrace{El(App(\omega, z))}^{\equiv : C},$$

for all $0 \leq j$, s.th. for all $a : A$, $b : B$, $c : \sum_{x:A} B \to \mathcal{U}_j$, $f : \prod_{x:A} \prod_{y:B} El(App(c, (x,y)))$,

$$\mathrm{ind}_{\sum_{x:A} B}(c, f, (x, y)) \equiv App(App(f, x), y) : \prod_{z : \sum_{x:A} B} C.$$

This is essentially the way Book-HoTT is presented (they furthermore choose to work with Russell-universes however).

Example 13: The dependent projections are defined for a type A with name $a: \mathcal{U}_i$, and a type family $\beta: A \to \mathcal{U}_i$, $x:A \vdash B :\equiv El((App_{\Pi}(\beta, x))\ type)$, as

$$\pi_1^{A,B} :\equiv ind_{\sum_{x:A} B}\left(\lambda\left(z:\sum_{x:A} B\right).\ a,\ \lambda(x:A).\lambda(y:B).\ x\right): \sum_{x:A} B \to A,$$

$$\pi_2^{A,B} :\equiv ind_{\sum_{x:A} B}\left(\lambda\left(z:\sum_{x:A} B\right).\ B[App(\pi_1^{A,B}, z)/x],\ \lambda(x:A).\lambda(y:B).\ y\right): \prod_{z:\sum_{x:A} B} B[\pi_1 z/x]$$

Notation 14: (1) If B is a constant type family over A (i.e. $\not\vdash A$ type), $\not\vdash B$ type, $x:A \vdash B$ type, resp. $b:\mathcal{U}_i$ with $\not\vdash El(b) \equiv B$ type and $\lambda(x:A).b: A \to \mathcal{U}_i$), one writes $A \times B :\equiv \sum_{x:A} B$.

2. Given types A, B, the type $A \simeq B$ of equivalences is defined as

$$A \simeq B :\equiv \sum_{f:A\to B}\left(\sum_{g:B\to A}(f\circ g =_{B\to B} 1_B) \times \sum_{h:B\to A}(h\circ f =_{A\to A} 1_A)\right)\ type$$

Lemma 15: Propositional Σ-uniqueness is provable, i.e. for A type, $x:A \vdash B$ type, the type

$$\prod_{t:\sum_{x:A} B} t =_{\sum_{x:A} B} (App(\pi_1, t), App(\pi_2, t))$$

is inhabited.

Proof: This is Corollary 2.7.3. in the HoTT-Book. The proof follows from a characterization of the identity types of a Σ-type as the Σ-type of associated identity-types of eliminators:

$$w : \prod_{t_1, t_2 : \sum_{x:A} B} \left((t_1 =_{\sum_{x:A} B} t_2) \simeq \left(\sum_{p : \pi_1(t_1) =_A \pi_1(t_2)} p_*(\pi_2(t_1)) =_{B[\pi_1(t_2)/x]} \pi_2(t_2) \right) \right)$$

RHS

p_*: "Transport of paths", see HoTT-Book

For $t_1 :\equiv t$, $t_2 :\equiv (App(\pi_1, t), App(\pi_2, t))$, we get

$\lambda(t_1 : \sum_{x:A} B).\ (refl_{\pi_1(t_1)}, refl_{\pi_2(t_1)}) : \prod_{f : \sum_{x:A} B} RHS$, and the statement follows by push-forward along w.

□

Exercise 16: 1. One can show propositional 1-Uniqueness as well: The type

$$\prod_{x:1} x = *$$

is inhabited.

(In other words, the terminal type 1 is "contractible". In fact, more is true:

There is a dependent function of equivalences of type $\prod_{x:1} \prod_{y:1} ((x =_* y) \simeq *)$

2. Similarly, $\prod_{x:0} \prod_{y:0} (x =_0 y) \simeq *$ is inhabited.

Propositional Π-Uniqueness (η-Congruence) is called Function Extensionality. It states that for A type, $x : A \to B$ type, the function

$$happly_{A \to B} : \prod_{f,g : A \to B} \left((f =_{A \to B} g) \to \prod_{x:A} App(f,x) =_B App(g,x) \right)$$

defined by Id-Elimination ("path-induction") on the canonical term

$$J(f : A \to B).\ J(x : A).\ \mathrm{refl}_{App(f,x)} : \prod_{f : A \to B} \prod_{x : A} App(f,x) =_B App(f,x)$$

is an equivalence (i.e. it can be extended to a term of type

$$\prod_{f,g : A \to B} \left((f =_{A \to B} g) \simeq \prod_{x : A} App(f,x) =_B App(g,x) \right)$$

whose pointwise first projection is $happly_{A,B}$.

Note that Function Extensionality as stated is a characterization of the identity-types of Π-types as Π-types of identity-types of associated eliminators.

These characterizations of compound type formers are crucial for mathematical practice, as without them it is virtually impossible to construct according non-trivial proofs of identity.

It is therefore interesting to note that Function Extensionality is not provable from the type theory set-up so far. It has to be imposed as an additional axiom.

It yet is of particular interest for various reasons, e.g. under Function Extensionality, inductively defined (dependent) functions from Elimination-principles given by fixed "boundary conditions" on canonical terms are unique up to propositional equality.

In summary, we have determined the identity-types of all type-formers up to equivalence of types except universes. A characterization of such identity-types is given by Voevodsky's Univalence Axiom.

Again by Id-Elimination, the map

$$\lambda(a:\mathcal{U}_i).(1_{E(a)}, (1_{f(a)}, \mathrm{refl}_{1_{f(a)}}), (1_{f(a)}, \mathrm{refl}_{1_{f(a)}})) : \prod_{a:\mathcal{U}_i} E(a) \simeq E(a)$$

induces a function

$$\mathrm{idtoequiv}_{\mathcal{U}_i} : \prod_{a:\mathcal{U}_i} \prod_{b:\mathcal{U}_i} (a =_{\mathcal{U}_i} b \longrightarrow E(a) \simeq E(b)) .$$

Definition 17: The universe $\mathcal{U}_i$ – or rather the type family $x:\mathcal{U}_i \vdash E(x)\ \mathrm{type}$ – is *univalent* if the function $\mathrm{idtoequiv}_{\mathcal{U}_i}$ can be extended to a dependent function of equivalences.

Remark 18: Univalence can be stated for any type family $x:A \vdash B\ \mathrm{type}$ requiring that the analogously defined function

$$\mathrm{idtoequiv}_{A\vert B} : \prod_{a_1:A} \prod_{a_2:A} (a_1 =_A a_2 \longrightarrow B[a_1/x] \simeq B[a_2/x])$$

can be extended accordingly.

Voevodsky's Univalence Axiom (UA) states that

$$\forall i \geq 0: \quad x : U_i \vdash E(i)(x) \text{ type is univalent,}$$

Consequences of the Univalence Axiom:

1. the propositional uniqueness priveiples hold up to propositional equality whenever they hold up to equivalence.

2. $\forall i \geq 0,\ a, b : U_i$:

$$\begin{array}{ccc} \alpha =_{U_i} \beta & \xrightarrow{e_i(\alpha,\beta)} & e(i(\alpha) =_{U_{i+1}} e(i(\beta) \\ \text{idtoequiv}_{U_i}(a,b) \downarrow & \textcircled{*} & \downarrow \text{idtoequiv}_{U_{i+1}}(e_i(a), e_i(b)) \\ E(i(\alpha) \simeq E(i(\beta) & \textcircled{\equiv} & E\ell_{i+1}(e_i(\alpha)) \simeq E\ell_{i+1}(e_i(\beta)) \end{array}$$

$\textcircled{*}$ up to propositional equality.

$$\Rightarrow \quad \alpha =_{U_i} \beta \xrightarrow[el_i(\alpha,\beta)]{\simeq} e(i(\alpha) =_{U_{i+1}} e(i(\beta) \ .$$

Corollary 19: Under UA, the maps $\lambda(a : U_i).\ e_i(a) : U_i \to U_{i+1}$ are "fully faithful".

(From ∞-grapoidal viewpoint of types.

3. Under UA, although the UIP holds for the type-formers $0, 1$ and their compounds, it fails for $\mathcal{U}_0$ if one adds coproduct types to the logical rules. (Indeed, given $\cdot \sqcup \cdot : \mathcal{U}_0$ s.th. $\mathsf{E}(\cdot \sqcup \cdot) \equiv 1+1$ type, one obtains

$$\Big((\cdot \sqcup \cdot) =_{\mathcal{U}_0} (\cdot \sqcup \cdot) \Big) \simeq \Big(1+1 \simeq 1+1 \Big)$$
$$\simeq 1+1 .$$

Thus, $\mathcal{U}_0$ has identity-types with provably non-identical identity-proofs.

4. The UA implies function Extensionality (see HoTT-Book, Section 4.9).

The fundamental fact that identity is an equivalence relation translates under proof-relevance to the following operations.

Exercise 20: Let A be a type. Then $\lambda(a:A).\, \mathrm{refl}_a : \prod_{a:A} a =_A a$. Furthermore,

1. There is a term $\mathrm{sym}_A : \prod_{a_1:A} \prod_{a_2:A} (a_1 =_A a_2 \longrightarrow a_2 =_A a_1)$

such that $\mathrm{sym}_A(a_1, a_1, \mathrm{refl}_{a_1}) \equiv \mathrm{refl}_{a_1} : a =_A a$ for all $a:A$.

2. There is a term $\mathrm{trans}_A : \prod_{a_1:A} \prod_{a_2:A} \prod_{a_3:A} a_1 =_A a_2 \longrightarrow (a_2 =_A a_3 \longrightarrow a_1 =_A a_3)$

such that - $\mathrm{trans}_A(a_1, a_1, b, \mathrm{refl}_{a_1}) \equiv 1_{a=_A b} : (a =_A b \longrightarrow a =_A b)$,

- $\lambda(p : a =_A b).\, \mathrm{trans}_A(a, b, b, p, \mathrm{refl}_b) \equiv 1_{a=_A b} : (a =_A b \longrightarrow a =_A b)$.

3. Leibniz' Law (Indiscernibility of identicals):

For all $i \geq 0$, there is a term

$$\mathrm{L}^{A}_{i} : \prod_{C : A \to \mathcal{U}_i} \prod_{a:A} \prod_{b:A} \left(a =_A b \to \left(\mathrm{El}(\mathrm{App}(C, a)) \simeq \mathrm{El}(\mathrm{App}(C, b)) \right) \right)$$

such that

$$\mathrm{L}^{A}_{i}(C, a, a, \mathrm{refl}_a) \equiv 1_{\mathrm{El}(\mathrm{App}(C,a))} : \mathrm{El}(\mathrm{App}(C,a)) \to \mathrm{El}(\mathrm{App}(C,a)).$$

Indeed, the dependent type formers we have introduced allow for a proof-relevant propositional interpretation in the spirit of the Curry-Howard-Lambek correspondence.

1st order logic	MLTT
$\bot$	0
$\top$	1
$\exists x \in A : \varphi(x)$	$\sum_{x:A} \varphi$
$\varphi \wedge \psi$	$\varphi \times \psi$
$\forall x \in A : \varphi(x)$	$\prod_{x:A} \varphi$
$\varphi \to \psi$	$\varphi \to \psi$
$s = t$ for $s, t \in A$	$s =_A t$

One may therefore study the corresponding logic which leads to a surprising interplay of classical axioms of symbolic logic such as the Law of Excluded Middle, the Axiom of Choice and others on the one hand, and homotopical structures associated to the UA on the other.