



Worksheet 2

HoTTEST Summer School 2022

The HoTTEST TAs, and
08 July 2022

1 (★)

The type **suit** is a type generated by four constructors:

Hearts : suit	Diamonds : suit
Clubs : suit	Spades : suit

Suppose we had a type family $s : \mathbf{suit} \vdash P(s)$ type. What data do we need to supply in order to define a term of the following type?

$$\prod_{s:\mathbf{suit}} P(s)$$

2 (★★)

Here's an informal description of a type:

$\mathbb{1}$ is a type generated by one constructor term, $\star : \mathbb{1}$.

Express this description formally as two inference rules, a 'Formation' rule, and an 'Introduction' rule (analogously to how we introduced **bool** in lecture).

Now, write the induction principle for $\mathbb{1}$, where $x : \mathbb{1} \vdash D$ type is some type family.

Instantiate this for the constant type family, i.e. D doesn't depend on $x : \mathbb{1}$ and is always a fixed type D . What do the elimination and computation rules for $\mathbb{1}$ say?

3 (★★)

Define a boolean-valued 'less than' operation on natural numbers:

$$<_2 \quad : \quad \mathbb{N} \rightarrow \mathbb{N} \rightarrow \mathbf{bool}$$

so that $(m <_2 n) \doteq \mathbf{true}$ when m is less than n , and \mathbf{false} otherwise.

Using this definition of `add`,

$$\begin{aligned} \mathbf{add} \ 0 \quad n &\doteq n \\ \mathbf{add} \ s(m) \ n &\doteq s(\mathbf{add} \ m \ n) \end{aligned}$$

Compute the term

$$(\mathbf{add} \ s(s(0)) \ s(s(0))) <_2 (\mathbf{add} \ s(0) \ s(0))$$

to either \mathbf{true} or \mathbf{false} .

4 (★★)

Fix some type X , and some type family $x : X, x' : X \vdash P(x, x')$ type. Write a term of type

$$\sum_{b:X} \prod_{a:X} P(a, b) \rightarrow \prod_{a:X} \sum_{b:X} P(a, b)$$

What does this say, under our logical interpretation?

5 (★ ★ ★)

\emptyset is the *empty type*: there are no terms of type \emptyset . It has the following induction principle:

For any type family $x : \emptyset \vdash Q(x)$ type, we have a term $\text{ind}_{\emptyset} : \prod_{x:\emptyset} Q(x)$.

What does this say when $Q(x)$ is a fixed type Q ?

Remember that we interpret types to be logical propositions, and terms/inhabitants to be proofs or witnesses of those propositions. What proposition does \emptyset represent?

If P is some proposition, what is the logical meaning of $P \rightarrow \emptyset$?

We write $\neg P$ as an abbreviation for $P \rightarrow \emptyset$

Write a term of type $\neg\neg\mathbb{1}$.

Is there a term of type $\neg\neg\emptyset$? Why or why not?

Let P and Q be types. We will write $P \leftrightarrow Q$ for the type $(P \rightarrow Q) \times (Q \rightarrow P)$. Use the fact that $\neg P$ is defined as the type $P \rightarrow \emptyset$ of functions from P to the empty type to give type theoretic proofs of the constructive tautologies

(i) $\neg(P \times \neg P)$

(ii) $\neg(P \leftrightarrow \neg P)$

