

# INTRODUCTION TO CATEGORICAL LOGIC

IVAN DI LIBERTI


*rules*

- Hand your exercises by the midnight of the **7th of May** via email. Make my life easier: include the word **CL23 in the subject**.
- Pick at least one exercise from each of the yellow groups.
- At each stage of the exercise sheet, you can give for granted the statements of all the exercises that come before the one you are solving.
- You must charge at least **3** batteries!


*Example.* The vector of exercises [1,5,8,12,15,18] would pass this sheet.

## EXERCISES



*doctrines and types*


**Exercise 1** (Naming, ) Consider the category of sets and the usual powerset doctrine defined over it,  $\mathcal{P} : \mathbf{Set}^\circ \rightarrow \mathbf{InfLat}$ . Using the usual *epi-mono* factorization, we can define a functor  $[-]_X : \mathbf{Set}_{/X} \rightarrow \mathcal{P}(X)$ . Inspired by this construction, for  $\mathcal{P} : \mathbf{C}^\circ \rightarrow \mathbf{InfLat}$  a doctrine with a sufficient amount of structure<sup>a</sup> construct a (pseudo)natural transformation

$$[-]_{(=)} : \mathbf{C}_{/(=)} \Rightarrow \mathcal{P}(=).$$

**Exercise 2** (...and necessity, ) We say that a doctrine has *comprehension schema* if the *naming* functor  $[-]_{(=)} : \mathbf{C}_{/A} \Rightarrow \mathcal{P}(A)$  of the exercise above has a right adjoint  $\{A : -\}$  for all  $A$ . Prove that if  $\mathcal{P} : \mathbf{C}^\circ \rightarrow \mathbf{InfLat}$  is a doctrine with a sufficient amount of structure, the canonical subobject doctrine defined on its category of elements has comprehension schema,

$$\mathbf{Sub} : \mathbf{Elts}(\mathcal{P})^\circ \rightarrow \mathbf{InfLat}.$$

**Exercise 3** (, ). Provide a type theoretic interpretation of a doctrine with comprehension schema.

**Exercise 4** () Provide a translation between the notion of comprehension category and that of category with display maps.

<sup>a</sup>It is enough that  $\mathbf{C}$  has finite limits and  $\mathcal{P}f$  has a left adjoint.

*topoi as spaces*

**Exercise 5** (□). Show that the category of sheaves over the Sierpinski space is a presheaf topos. Which one?

**Exercise 6** (□). Show that  $\mathbf{Set}^{\rightarrow}$  has a closed subtopos and an open subtopos. Please, provide a full proof that the geometric morphisms you present have the property we require, don't just state it.

**Exercise 7** (□). Let  $X$  be a compact Hausdorff space. Show that the direct image of the terminal geometric morphism  $\Gamma_* : \mathbf{Sh}(X) \rightarrow \mathbf{Set}$  preserve directed colimits of monomorphisms.

*topoi as objects*

**Exercise 8** (□, ▢). Show that the bicategory of topoi has (pseudo)colimits. *Hint.* This exercise is not as hard as it may seem.

**Exercise 9** (□, ▢). Show that the bicategory of topoi has (pseudo)pullbacks. *Hint.* Yes, this exercise is too hard.

**Exercise 10** (□). Show that open geometric morphisms are pullback stable.

**Exercise 11** (□). Show that closed geometric morphisms are pullback stable.

*topoi as objects*

**Exercise 12** (□, ▢). Show that there is a bijective correspondence between

$$\mathbf{Sub}_{\mathcal{E}}(1) \simeq \mathbf{Topoi}(\mathcal{E}, \mathbf{Set}^{\rightarrow}).$$

**Exercise 13** (□). Show that the bicategory of topoi has a classifier of closed embeddings, i.e., there exists a closed embedding  $p : \mathcal{F}_1 \rightarrow \mathcal{F}_2$  such that every closed subtopos can be obtained by pulling back a geometric morphism along  $p$ .

$$\begin{array}{ccc} \bullet & \dashrightarrow & \mathcal{F}_1 \\ \downarrow & & \downarrow p \\ \mathcal{E} & \longrightarrow & \mathcal{F}_2 \end{array}$$

Prove an analogous statement also for open embeddings. *Hint.* To get the proper intuition, first solve it for spaces, then for locales, and then for topoi.

**Exercise 14** (□). Show that every open subtopos is *complemented*, i.e. there exists a closed subtopos that is its complement in the lattice of subtopoi.

*topoi as sets*

**Exercise 15** (▣). Provide a complete description of the subobject classifier in  $\mathbf{Set}^{\mathbb{N}}$ , where the category structure of  $\mathbb{N}$  is the expected posetal one.

**Exercise 16** (▣). Show that every topos has a partial map classifier for every object. *Hint:* What are the partial map classifiers in  $\mathbf{Set}$ ?

**Exercise 17** (▣). Prove that an object of a topos  $\mathcal{E}$  is injective (with respect to monos) if and only if it is a retract of  $\Omega^x$  for some  $x$ . Deduce that if  $e$  is injective then the functor  $[-, e] : \mathcal{E}^{\circ} \rightarrow \mathcal{E}$  preserves reflexive coequalizers.

*topoi as theories*

**Exercise 18** (▣). Consider the category of non empty finite sets  $\mathbf{Fin}_{>0}$ . What theory does  $\mathbf{Set}^{\mathbf{Fin}_{>0}}$  classify?

**Exercise 19** (▣). Consider the category of finite sets and monomorphisms  $\mathbf{Fin}_{\hookrightarrow}$ . What theory does  $\mathbf{Set}^{\mathbf{Fin}_{\hookrightarrow}}$  classify?

**Exercise 20** (▣). Consider the category of finite sets and epimorphisms  $\mathbf{Fin}_{\twoheadrightarrow}$ . What theory does  $\mathbf{Set}^{\mathbf{Fin}_{\twoheadrightarrow}}$  classify?

**Exercise 21** (▣). Consider the category of pointed finite sets  $\mathbf{Fin}_{\bullet}$ . What theory does  $\mathbf{Set}^{\mathbf{Fin}_{\bullet}}$  classify?

**Exercise 22** (▣). Consider the comma topos below, and assume comma topos exist in the bicategory of topoi. Can you describe how does a **Set**-model of the comma topos look like (in terms of models of  $\mathbb{T}_1$  and  $\mathbb{T}_2$ )?

$$\begin{array}{ccc}
 \bullet & \xrightarrow{\quad\quad} & \mathbf{Set}[\mathbb{T}_1] \\
 \downarrow & \searrow \lambda & \downarrow \\
 \mathbf{Set}[\mathbb{T}_2] & \xrightarrow{\quad\quad} & \mathbf{Set}
 \end{array}$$

Feel free to assume that  $\mathbb{T}_1$  and  $\mathbb{T}_2$  are single sorted if you wish.

**The riddle** (▲). Show that a presheaf topos  $\mathbf{Set}^{\mathbf{C}}$  is boolean if and only if  $\mathbf{C}$  is a groupoid.