Exam 2021-2022

Khalil Ghorbal

January 2022

Pick up at least two out of the three proposed problems below according to your taste. If you enjoyed the mindset of Category theory, check out Problem 1, it has it all. If, however, you don't really care whether every monad arises from an adjunction, Problem 2 might be more suited as it manipulates deductive systems: all you need is a firm logician hat. Finally, if you are more language-oriented, you can chew on the (untyped) constructions of Problem 3. This being said, keep in mind that the three suggested problems are part of the *same story*. The more you appreciate this fact, the more enlightened and powerful you will be, regardless of the next steps of your curriculum.

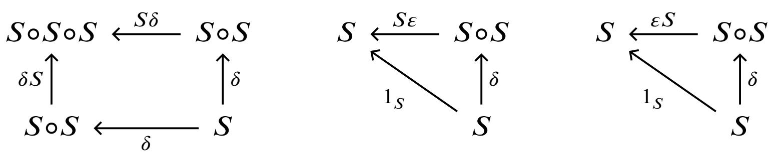
Stay focused. Clear your thoughts. Enjoy the dive.

1 Functional Completeness as a Universality Property

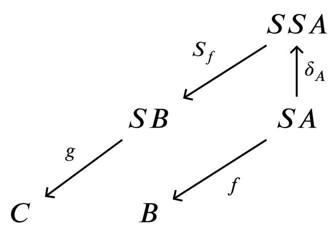
A *comonad* on a category $\mathscr{A}$ is a monad in the opposite category $\mathscr{A}^{op}$, that is a cotriple (S, ε, δ), where $S : \mathscr{A} \to \mathscr{A}$ is a functor equipped with a counit and a co-multiplication satisfying the associativity and identity laws (needless to say that ε and δ are natural transformations). As we did for monads, we can define the Kleisli category $\mathscr{A}_S$ of a comonad (S, ε, δ) on $\mathscr{A}$ with morphisms $A \to B$ in $\mathscr{A}_S$ whenever $SA \to B$ is a morphism in $\mathscr{A}$ (everything else is like we've seen but with inverting the arrows).

1. State explicitly the associativity and identity laws for a comonad (three diagrams are expected.)
2. State explicitly the identity arrows and composition of morphisms in $\mathscr{A}_S$ (a diagram is expected for the composition).

1.sol



2.sol The identity arrow $1_A : A \twoheadrightarrow A$ is defined as $\varepsilon_A : SA \to A$. Two morphisms $f : A \twoheadrightarrow B$ and $g : B \twoheadrightarrow C$ are composed as $g \circ S_f \circ \delta_A$:



Let $\mathscr{A}$ be a Cartesian category where $\pi_{A,B}$ and $\pi'_{A,B}$ denote the projections out of the product $A \times B$ of $A, B \in \mathscr{A}$. We use $\langle f, g\rangle$ to denote the *pairing* of f and g, that is the unique map $C \to A \times B$ where $f : C \to A$ and $g : C \to B$.

For any object A in $\mathscr{A}$, define $S_A := A\times -$, $\quad \varepsilon_A(B) := \pi'_{A,B}$, $\quad \delta_A(B) := \langle \pi_{A,B}, 1_{A\times B}\rangle$ (for clarity, we used $\varepsilon_A(B)$ and $\delta_A(B)$ instead of $(\varepsilon_A)_B$ and $(\delta_A)_B$ to avoid double subscripts).

3. Show that $(S_A, \varepsilon_A, \delta_A)$ defines a cotriple of $\mathscr{A}$. We will denote $\mathscr{A}_{S_A}$, or simply $\mathscr{A}_A$, the Kleisli category of the comonad $(S_A, \varepsilon_A, \delta_A)$ on $\mathscr{A}$.

3.sol A is a fixed object of $\mathscr{A}$. S_A is a functor, so it is also defined on arrows of A. Let $f : B \to C$, then $S_A f = \langle \pi_{A,B}, f \circ \pi'_{A,B}\rangle$. We check that ε_A is a counit, that is a natural transformation $S \to 1_{\mathscr{A}}$. Let $B, C \in \mathscr{A}$, $f : B \to C$, by construction of the pairing $\pi'_{A,C} \circ S_A f = f \circ \pi'_{A,B}$, thus the following diagram commutes

$$\begin{array}{ccc} S_A B & \xrightarrow{S_A f} & S_A C \\ {\scriptstyle \varepsilon_A(B)}\downarrow & & \downarrow{\scriptstyle \varepsilon_A(C)} \\ B & \xrightarrow[f]{} & C \end{array} \quad = \quad \begin{array}{ccc} A \times B & \xrightarrow{S_A f} & A \times C \\ {\scriptstyle \pi'_{A,B}}\downarrow & & \downarrow{\scriptstyle \pi'_{A,C}} \\ B & \xrightarrow[f]{} & C \end{array}$$

and since $S_A S_A f = \langle \pi_{A,A\times B}, (S_A f) \circ \pi'_{A,A\times B}\rangle$, the following diagram commutes as well

$$\begin{array}{ccc} S_A B & \xrightarrow{S_A f} & S_A C \\ {\scriptstyle \delta_A(B)}\downarrow & & \downarrow{\scriptstyle \delta_A(C)} \\ S_A S_A B & \xrightarrow[S_A S_A f]{} & S_A S_A C \end{array}$$

Let $\mathcal{A}[x]$ denote the polynomial category defined over $\mathcal{A}$ assuming the undetermined $x : A_0 \to A$. Let $H_x : \mathcal{A} \to \mathcal{A}[x]$ denote the Cartesian functor that sends $f : A \to B$ onto the constant polynomial with the same name in $\mathcal{A}[x]$ (in words, H_x defines a trivial injection that regards constants as polynomials). The functional completeness can be rephrased as the following *universality property* for H_x. Given a Cartesian category $\mathcal{A}$, any Cartesian functor $F : \mathcal{A} \to \mathcal{B}$ and any arrow $y : F(A_0) \to F(A)$ in $\mathcal{B}$, there exists a unique Cartesian functor $F' : \mathcal{A}[x] \to \mathcal{B}$ such that $F'(x) = y$ and $F' \circ H_x = F'H_x = F$. (The proof of this statement is very similar to—if not the same as—the proof of the deductive theorem deconstructed in Section 2). We will use this universality property to show (with elegance) that the polynomial category $\mathcal{A}[x]$ is isomorphic to the Kleisly category $\mathcal{A}_A$ (A and x are indeed related since A is the codomain of x).

4. Show that $\mathcal{A}_A$ is a Cartesian category. (You need to show that a product exists in $\mathcal{A}_A$, and a product is not a mere isolated object, that is you need to explicit the projections and maps to the terminal object—as a special case of the product of zero elements).

4.sol $\mathcal{A}$ is a Cartesian category, thus there is already a product $B \times C$. All we need to exhibit are the projections in $\mathcal{A}_A$ from $B \times C$ to B and C. The natural candidates are $\pi_{B,C} \circ \varepsilon_A(B \times C)$ and $\pi'_{B,C} \circ \varepsilon_A(B \times C)$. The unique map in $\mathcal{A}_A$ from $B \times C$ to 1, the terminal object of $\mathcal{A}$ would be the unique map in $\mathcal{A}$ from $A \times B \times C$ to 1. Thus $\mathcal{A}_A$ is a Cartesian category.

Define the functor $H_A : \mathcal{A} \to \mathcal{A}_A$ by $H_A(B) = B$ and $H_A(f) = f\pi'_{A,C}$ for objects B and arrows $f : C \to B$.

5. Check that H_A is a Cartesian functor.

6. Assume that H_A enjoys the same universality property of H_x with $\pi_{A,1}$ as the undetermined x. Prove that $\mathcal{A}[x]$ is isomorphic to $\mathcal{A}_A$. (Even if you know nothing about polynomial categories and monads, you should be able to prove this just by exploiting universality.)

5.sol We need to check that H_A preserves the Cartesian structure. It's clear that $H_A(B \times C) = H_A(B) \times H_A(C)$, we check below that projection maps and the pairing are preserved as expected. For the map (and objects) $\pi_{B,C} : B \times C \to B$, one gets $H_A(\pi_{B,C}) : H_A(B \times C) \twoheadrightarrow H_A(B)$, or equivalently $\pi_{B,C}\pi'_{A,B\times C} : A \times B \times C \to B$ which is equal to $\pi_{B,C}$ in $\mathcal{A}_A$. For pairing, we need to show that a pairing $\langle f, g \rangle : D \to B \times C$ is transformed via H_A to the pairing $\langle H_A(f), H_A(g) \rangle$ in $\mathcal{A}_A$ or equivalently to $\langle f\pi'_{A,D}, g\pi'_{A,D} \rangle$ in $\mathcal{A}$ which holds by definition of H_A.

6.sol To prove that $\mathcal{A}[x]$ is isomorphic to $\mathcal{A}_A$ we use twice the universality property as follows. First use $\mathcal{A}_A$ as $\mathcal{B}$ and H_A as F, then there exists a unique functor $H'_A : \mathcal{A}[x] \to \mathcal{A}_A$ such that $H'_A H_x = H_A$. Then use $\mathcal{A}[x]$ as $\mathcal{B}$ and H_x as F, then there exists a unique functor $H'_x : \mathcal{A}_A \to \mathcal{A}[x]$ such that $H'_x H_A = H_x$. Thus $H'_A H'_x H_A = H'_A H_x = H_A$ and $H'_x H'_A H_x = H'_x H_A = H_x$. But then the functors $1_{\mathcal{A}_A}$ and $1_{\mathcal{A}[x]}$ satisfy respectively $1_{\mathcal{A}_A} H_A = H_A$ and $1_{\mathcal{A}[x]} H_x = H_x$. By uniqueness $H'_A H'_x = 1_{\mathcal{A}_A}$ and $H'_x H'_A = 1_{\mathcal{A}[x]}$. In short, two objects that verify the same universality property are isomorphic.

Bonus (prove that H_A has indeeded the assumed universality property). Let $F : \mathcal{A} \to \mathcal{B}$ denote a Cartesian functor and $y : 1 \to F(A)$ a given arrow in $\mathcal{B}$. We show constructively the existence of a unique Cartesian functor $F' : \mathcal{A}_A \to \mathcal{B}$ that satisfies the desired properties, that is $F'H_A = F$ and $F'(\pi_{A,a}) = y$. Let F' be defined on objects and arrows as $F'(B) = FB$ and $F'(f) = F(f)\langle yF(B)\bullet, 1_{F(B)} \rangle$, where $F(B)\bullet : F(B) \to 1$.

7a. Check that F' is Cartesian.

7b. Check that it satisfies the desired properties.

7c. Prove uniqueness.

2 Deduction Theorem

The standard (and simpler) form of the *deduction theorem* asserts that if $A \wedge B \to C$ then $A \to (C \Leftarrow B)$ (you probably already encountered a similar statement where arrows are denoted by $\vdash$, reads "entails"). However, as soon as one adjoins an assumption $x : \mathbf{T} \to A$ (that is a proof x for the formula A), one obtains a new deductive system $\mathscr{D}(x)$ on which we stated the general form of the deduction theorem. In what follows, you will be guided to prove the theorem for positive intuitionistic propositional calculus. (This is a very general scheme for many proofs in formal languages and abstract algebras.)

1. You may (or may not!) have noticed a circular argument in introducing the 'if' operator $\Leftarrow$ since we also used a sort of 'if ... then ... ' construction (via the inference rule) to introduce $\Leftarrow$. (This is sometimes called the Zeno paradox of logic). How did we solve this issue?

2. What are the (three) primitive operators on proofs in $\mathscr{D}(x)$?

3. Let $\varphi(x)$ denote a proof $B \to C$ in $\mathscr{D}(x)$ where $x : \mathbf{T} \to A$. Deconstruct $\varphi(x)$ using pairing and do the same for transposition. (Bonus: make explicit the five possible forms for $\varphi(x)$).

4. For pairing and transposition, construct an explicit proof $f(x) : A \wedge B \to C$. Here is an example: if $p : B \to C$ is a proof in $\mathscr{D}$ (which is therefore independent from x), then $f = p \circ \pi'_{A,B}$ (which is also independent from x). We can also write f using the λ-abstraction $\lambda_{x:A} p$. Feel free to use similar notations.

5. Observe that $f(x)$, through the primitive operators, deconstructs $\varphi(x)$ into *shorter* proofs. We can make this intuition precise by defining an inductive notion of length on proofs: for instance if $p : B \to C$ exists already in $\mathscr{D}$, then $\varphi(x)$ has length 0 (the constant polynomial). What is the length of a proof pairing two proofs of lengths $n, m \geq 0$?

6. Sketch a proof by induction for the deduction theorem (in $\mathscr{D}(x)$) (outline the main steps based on your previous answers).

7. Bonus: What is the missing ingredient in order to state *functional completeness* in the corresponding Cartesian closed category of $\mathscr{D}(x)$?

1.sol Both constructions, while similar, do not operate at the same level. The 'if' operator decorates objects (propositions) of the deductive system (you can think of these objects as knowledge), whereas the inference rules are the proofs of those propositions, so as if we are encoding, within the same system, the fact that we can prove something. The inference rule is therefore regarded as the meta-level of knowledge, or how to saturate the system with new facts from known ones. This is very similar of talking about the set of functions between two sets as a set itself!

2.sol pairing, conjunction (or product) and transposition.

3.sol The five forms are (i) Constant. $\varphi(x) = k$ where $k : B \to C$ is (already) a proof in $\mathscr{D}$, (ii) Variable. $\varphi(x) = x$ with $B = \mathbf{T}$ and $C = A$, (iii) Pairing. $\varphi(x) = \langle \psi(x), \xi(x) \rangle$ where $\psi(x) : B \to C'$ and $\xi(x) : B \to C''$, $C = C' \wedge C''$, (iv) Product. $\varphi(x) = \psi(x)\xi(x)$, where $\psi(x) : B \to D$ and $\xi(x) : D \to C$, and finally (v) Transposition. $\varphi(x) = \bar{\xi}(x)$, where $\xi(x) : B \wedge C' \to C''$, and $C = C'' \Leftarrow C'$

4.sol For pairing (case (iii) above), it suffices to construct $\psi(x)\pi'_{A,B} : A \wedge B \to C'$ and $\xi(x)\pi'_{A,B} : A \wedge B \to C''$, so that the pairing $f(x) = \langle \psi(x)\pi'_{A,B}, \xi(x)\pi'_{A,B} \rangle$ gives a proof $A \wedge B \to C' \wedge C''$ (recall that $C = C' \wedge C''$). For the product (case (iv) above), $f(x) = \psi(x)\xi(x)\pi'_{A,B}$. Finally, for transposition (case (v) above), $f(x) = \overline{\xi(x)\pi'_{A,B \wedge C'}}$.

5.sol The length of a proof can be defined inductively: the length is zero for cases (i) and (ii), it is the sum of the lengths of $\xi(x)$ and $\psi(x)$ plus 1 in cases (iii) and (iv) and the length of $\xi(x)$ plus 1 in case (v).

6.sol The proof is by induction on the length of the proof. One starts by $\phi(x)$, and decompose it following one of the 5 cases detailed above, then in each subcase, the proof decomposes again, at each step the length is strictly decreasing; thus the process ends after finitely many steps and the final proof has no x in it.

3 Church's Numerals

Recall that the untyped λ-calculus is defined inductively by $t ::= x \mid t \wr t' \mid \lambda_x.t$. For simplicity, we will use concatenation to encode $\wr$, so that we write tt' for $t \wr t'$. Church defined some sort of natural numbers, called *numerals*, using the untyped λ-calculus as follows. First, he introduced the operator $\star$ on λ-terms $t \star t' := \lambda_x.(t(t'x))$ .. Then, he went on defining

$$0 := \lambda_x.(\lambda_x.x), \quad 1 := \lambda_x.x, \quad 2 := \lambda_x.(x \star x), \dots$$

so that $1f = f$, $2f = f \star f$, etc. as if a numeral n encodes the process of applying the $\star$ operator n times to its argument f.

1. What is $0f$? Is it expected? Comment.
2. What does the $\star$ operator encode as a standard operation on natural numbers?
3. What does $\lambda_x(x \star (nx))$ represent for a numeral n? (Hint: think of the basic ingredients you need to define a natural number object).
4. How can you encode exponentiation of natural numbers?
5. Bonus: Write the sum of two numerals as a λ-term.

1.sol Applying 0 to f gives $\lambda_x.x$ which is 1. Here numerals are defined using a binary operator, the $\star$. Recall that the product of zero operands has a sense and corresponds to the terminal object, often denoted 1... what a coincidence! At this point, you should already have guessed the answer to the next question.

2.sol The star operator is exactly the multiplication, indeed we can define $n \times m$ as $n \star m$.

3.sol $\lambda_x(x \star (nx))$ encodes precisely the successor of the numeral n, which is indeed essential in any natural numbers system.

4.sol Using concatenation, one encodes m^n as nm (the latter is not the product of m and n, but the concatenation in the λ-calculus; We could have written it as $n \wr m$).

So you might think that, since numerals behave like natural numbers, there is a built-in 'type' (precisely the one for natural numbers) that comes for free in an untyped λ-calculus... let's push this further yet before concluding.

1. Suppose that x has type A in the definition of $\star$, what would be the type of n and m in $n \star m$?
2. Is this type coherent with the product of two numerals? What about exponentiation of numerals?
3. State under which extra condition on the type A all numerals would have the same type. (So after all, numerals aren't really natural numbers...)

1.sol If x has type A, then $(n \star m)x = n(mx)$, since we expect n and m to have the same type, both n and m must be typed A^A. Denote A^A by B.

2.sol If m is typed B, then for nm to make sense, n must be typed B^B. Indeed, the concatenation encodes the exponentiation of numerals and should therefore has the same type B.

3.sol We thus end up requiring B to be the same as B^B. But this is clearly not true in general. Numerals cannot have the same type unless, they are typed with a special type satisfying $B^B = B$.