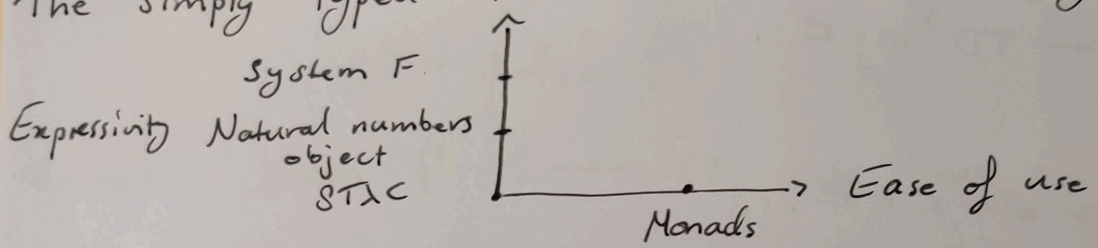
Notions of computation

What is a program?

Easy: a term in the simply typed λ-calculus.

... Is it though?

The simply typed λ-calculus is relatively weak.



Orthogonal to this, is the ease of use of a program, or the ease of "reasoning" about a program.

Moggi made the observation that identifying programs with input / output behaviour is too restrictive, as programs are more than that. Programs may differ in efficiency, resource consumption, side effects, etc.

At the same time though, identifying programs with functions is mathematically (and intuitively) very convenient.

So can we strike a balance?

Answer: yes, using <u>monads</u>.

Def^{n}: A <u>Kleisli triple</u> over a category $\mathcal{C}$ is a triple $(T, \eta, -^{*})$ consisting of:

- An association of an object $TC \in \mathcal{C}$ to every object $C \in \mathcal{C}$,
- A collection of functions $\eta = \{\eta_C : C \longrightarrow TC\}_{C \in \mathcal{C}}$,
- An association of a morphism $f^{*}: TA \longrightarrow TB$ to every morphism $f : A \longrightarrow TB$ in $\mathcal{C}$.

satisfying:

- $\eta_C^{*} = id_{TC}$ for all $C \in \mathcal{C}$,
- $f^{*} \circ \eta_C = f$, for all $f : C \longrightarrow TB$,
- $(g^{*} \circ f^{*}) = (g^{*} \circ f)^{*}$, whenever such a composition makes sense.

We will see later that there is a bijection between Kleisli triples and monads.

Eg) Let $\mathcal{C} = \underline{Set}$ and:

- $TC := C \amalg \{\perp\}$,
- $\eta_C := C \longrightarrow C \amalg \{\perp\}$ (canonical inclusion),
- given $f : A \longrightarrow TB$:

$$f^{*} : TA \longrightarrow TB$$
$$x \longmapsto \begin{cases} f(x), & x \in A \\ \perp, & \text{else}. \end{cases}$$

This is a model of a program which may fail to execute. We prove this is a Kleisli triple:

any $C \in \underline{\text{Set}}$:

$$\eta_C : C \longrightarrow TC$$
$$x \longmapsto x$$

so
$$\eta_C^* : TC \longrightarrow TC$$
$$x \longmapsto \begin{cases} x, & x \in C \\ \perp, & \text{else} \end{cases}$$

this is the identity function on TC, so

$$\eta_C^* = Id_{TC}.$$

Moreover, given $f : C \longrightarrow TB$, we have

$$C \xrightarrow{\eta_C} TC \xrightarrow{f^*} TB$$
$$x \longmapsto x \longmapsto \begin{cases} f(x), & x \in C \\ \perp, & \text{else} \end{cases}$$

But $x \in C$, so $(f^* \circ \eta_C)(x) = f(x)$ $\forall x \in C$.

That is, $f^* \circ \eta_C = f$.

Lastly:
$(g^* \circ f^*)(x)$:

$$TA \xrightarrow{f^*} TB \xrightarrow{g^*} TC$$
$$x \longmapsto \begin{cases} f(x), & x \in A, \\ \perp, & \text{else} \end{cases} \longmapsto \begin{cases} g(f(x)), & x \in A \\ \perp, & \text{else} \end{cases}$$

on the other hand,

$$A \xrightarrow{f} TB \xrightarrow{g^*} TC$$
$$x \longmapsto f(x) \longmapsto g(f(x)),$$ so

$$(g^* \circ f)^*(x) = \begin{cases} g(f(x)), & x \in A \\ \perp, & \text{else}. \end{cases}$$

so $(g^* \circ f)^* = g^* \circ f^*$.

We can now understand the definition of a Kleisli triple better:

The $-^*$ operator encodes how to upgrade a morphism to carry the notion of computation under consideration.

In our example, the notion of computation was "failure", and given $f: A \longrightarrow TB$ which might fail, $f^*: TA \longrightarrow TB$ "carries the failure".

The morphisms $\{\eta_c : C \longrightarrow TC\}_{C \in \mathcal{C}}$ "embed" C into its counterpart which is conscious of the notion of computation. The word "embed" is in inverted commas because there is no necessity for η_c to be monic (but it tends to be).

Thus, η_c^* is an embedding adapted to allow for the notion of computation, this ought to be the identity on TC. In other words,

$$\eta_c^* = Id_{TC}.$$

Moreover, f^* restricted to the embedded part should just be f, that is:

$$f^* \circ \eta_c = f.$$

The final axiom is a coherence between composition and $-^*$, since $-^*$ adapts the domain of a phism, we can either do this before or after a composition, asking for these to be the same yields

$$(g^* \circ f^*) = (g^* \circ f)^* .$$

Theorem: There is a bijection between Kleisli triples over $\mathcal{C}$ and monads on $\mathcal{C}$.

Proof: Let $(T, \eta, -^*)$ be a Kleisli triple. Extend T to morphisms by:

$$T(f : A \to B) = (\eta_B f)^* : TA \to TB$$

and for any $C \in \mathcal{C}$ let

$$\mu_C = id_{TC}^* : TTC \to TC .$$

Naturality of η: let $f : A \to B$ be a morphism in $\mathcal{C}$,

We need:

$$\begin{tikzcd} A \arrow[r, "\eta_A"] \arrow[d, "f"'] & TA \arrow[d, "Tf = (\eta_B f)^*"] \\ B \arrow[r, "\eta_B"'] & TB \end{tikzcd}$$

to commute.

So we need $(\eta_B f)^* \eta_A = \eta_B f$

this holds by axiom 2 of Kleisli triples.

Naturality of μ: Need commutativity of:

$$\begin{tikzcd} TTA \arrow[r, "\mu_A = id_{TA}^*"] \arrow[d, "TTf = (\eta_{TB}(\eta_B f)^*)^*"'] & TA \arrow[d, "Tf = (\eta_B f)^*"] \\ TTB \arrow[r, "\mu_B = id_{TB}^*"'] & TB \end{tikzcd}$$

So we need:

$$(\eta_B f)^* \, \mathrm{Id}_{TA}^* = \mathrm{Id}_{TB}^* \, (\eta_{TB} (\eta_B f)^*)^*$$

We have:

$$\begin{aligned}(\eta_B f)^* \, \mathrm{Id}_{TA}^* &= ((\eta_B f)^* \, \mathrm{Id}_{TA})^* && \text{(Axiom 3)}\\ &= (\eta_B f)^{**}\end{aligned}$$

On the other hand:

$$\begin{aligned}&\mathrm{Id}_{TB}^* \, (\eta_{TB} (\eta_B f)^*)^*\\ &= (\mathrm{Id}_{TB}^* (\eta_{TB} (\eta_B f)^*))^* && \text{(Axiom 3)}\\ &= (\mathrm{Id}_{TB} (\eta_B f)^*)^* && \text{(Axiom 2)}\\ &= (\eta_B f)^{**}\end{aligned}$$

<u>Monadicity</u>:

Commutativity of:

$$\begin{tikzcd} T^3C \arrow[r, "\mu_{TC} = \mathrm{Id}_{TTC}^*"] \arrow[d, "T\mu_C = (\eta_C \mathrm{Id}_{TC}^*)^*"'] & T^2C \arrow[d, "\mu_C = \mathrm{Id}_{TC}^*"] \\ T^2C \arrow[r, "\mu_C"'] & TC \end{tikzcd}$$

Need: $\mathrm{Id}_{TC}^* (\eta_C \mathrm{Id}_{TC}^*)^* = \mathrm{Id}_{TC}^* \, \mathrm{Id}_{TTC}^*$

We have: $\mathrm{Id}_{TC}^* (\eta_C \mathrm{Id}_{TC}^*)^*$

$$\begin{aligned}&= (\mathrm{Id}_{TC}^* (\eta_C \mathrm{Id}_{TC}^*))^*\\ &= (\mathrm{Id}_{TC} \, \mathrm{Id}_{TC}^*)^*\\ &= \mathrm{Id}_{TC}^{**}\end{aligned}$$

the other hand:

$Id_{TC}{}^{*} \, Id_{TTC}{}^{*}$

$= (Id_{TC}{}^{*} \, Id_{TTC})^{*}$

$= Id_{TC}{}^{**}$

Commutativity of:

$$\begin{tikzcd}
TC \arrow[r, "T\eta_C"] \arrow[dr, "Id_{TC}"'] & T^2C \arrow[d, "\mu_C"] & TC \arrow[l, "\eta_{TC}"'] \arrow[dl, "Id_{TC}"] \\
& TC &
\end{tikzcd}$$

Need: $Id_{TC}{}^{*} (\eta_{TC}\eta_C)^{*} = Id_{TC}$ ①

and $Id_{TC}{}^{*} \, \eta_{TC} = Id_{TC}$. ②

①: $Id_{TC}{}^{*} (\eta_{TC}\eta_C)^{*}$

$= (Id_{TC}{}^{*} \eta_{TC} \eta_C)^{*}$

$= (Id_{TC} \eta_C)^{*}$

$= \eta_C{}^{*}$

$= Id_{TC}$ (Axiom 1).

②: Follows immediately from (Axiom 1) and (Axiom 2).

Conversely, given a monad (T, η, μ), define a Kleisli triple by: $f^{*} = \mu_B (Tf)$. $(f: A \to B)$

Then: $\eta_C{}^{*} = \mu_C (T\eta) = Id_{TC}$.

$$
\begin{aligned}
f^* \circ \eta_A &= \mu_B T f \eta_A \\
&= \mu_B \eta_{TA} f \\
&= f .
\end{aligned}
$$

Lastly, $(g^* \circ f^*)$

$$= \mu_C Tg \, \mu_B Tf$$

$$
\begin{tikzcd}
TA \arrow[r, "Tf"] & T^2B \arrow[r, "T^2g"] \arrow[d, "\mu_B"'] \arrow[dr, phantom, "\text{Need } \circlearrowleft"] & T^3C \arrow[d, "T\mu_C"] & \\
& TB \arrow[r, "Tg"'] & T^2C \arrow[r, "\mu_C"'] & TC
\end{tikzcd}
$$

We have by naturality of μ that

$$
\begin{tikzcd}
T^2B \arrow[r, "T^2g"] \arrow[d, "\mu_B"'] & T^3C \arrow[d, "\mu_{TC}"] \\
TB \arrow[r, "Tg"] & T^2C
\end{tikzcd}
$$

commutes, so we just need the compositions

$$
\begin{tikzcd}
T^2C \arrow[r, shift left, "T\mu_C"] \arrow[r, shift right, "\mu_{TC}"'] & T^2C \arrow[r, "\mu_C"] & TC
\end{tikzcd}
$$

to be equal, this follows from monadicity of μ.

<u>Bijection:</u>

Given a Kleisli triple $(T, \eta, -^*)$, we have a monad (T, η, μ) and then a Kleisli triple again $(T, \eta, -^\wedge)$ where

$$
\begin{aligned}
f^\wedge &= \mu_B Tf \\
&= {Id_{TB}}^* (\eta_B f)^* \\
&= ({Id_{TB}}^* \eta_B f)^* \\
&= f^* .
\end{aligned}
$$

Conversely, given a monad (T, η, μ) we have a Kleisli triple $(T, \eta, -^*)$ and then a monad again (S, η, ξ) where for $f : A \to B$:

$$
\begin{aligned}
S(f) = (\eta_B f)^* &= \mu_{TB} T(\eta_B f) \\
&= \mu_{TB} T(\eta_B) T(f) \\
&= T(f)
\end{aligned}
$$

$$\xi_c = \mathrm{Id}_{Tc}^* = \mu_c(T\mathrm{Id}_{Tc})$$
$$= \mu_c(\mathrm{Id}_{TTc})$$
$$= \mu_c.$$

$\square$

A more involved example: $\mathcal{C} = \underline{\text{Set}}$, and fix $S \in \mathcal{C}$.

Let $TA = (A \times S)^S$, and given $f : A \longrightarrow TB$ define:

$$f^* : (A \times S)^S \longrightarrow (B \times S)^S$$
$$g \longmapsto (f\pi_1, \pi_2)g$$

Assume also that S has a special state $\mathbb{1}$.

Then $\eta_A : A \longrightarrow (A \times S)^S$
$$a \longmapsto (s \mapsto (a, \mathbb{1}))$$

Exercise: prove this is a Kleisli triple.

This kleisli triple corresponds to the notion of computation of "state dependence".

Combining notions of computation:

Say T, S are monads on $\mathcal{C}$, then is TS a monad?

$$\mu_c^{TS} : TSTS(c) \longrightarrow TS(c)$$

Have $\mu_c^T : TT(c) \longrightarrow T(c)$
$$\mu_c^S : SS(c) \longrightarrow S(c)$$

So we need a natural transformation

$$ST \longrightarrow TS.$$

In general, there is no canonical way of doing this.

So monads don't compose in general.

But we can compose monads which admit a distributive law:

Def^n: Given monads (S, μ^S, η^S), (T, μ^T, η^T), a <u>distributive law</u> is a natural transformation

$$\chi : TS \longrightarrow ST$$

s.t. the diagrams:

$$\begin{tikzcd} TSS \arrow[r, "\chi_S"] \arrow[d, "T\mu^S"'] & STS \arrow[r, "S\chi"] & SST \arrow[d, "\mu^S_T"] \\ TS \arrow[rr, "\chi"'] & & ST \end{tikzcd} \qquad \begin{tikzcd} & T \arrow[dl, "T\eta^S"'] \arrow[dr, "\eta^S_T"] & \\ TS \arrow[rr, "\chi"'] & & ST \end{tikzcd}$$

$$\begin{tikzcd} TTS \arrow[r, "T\chi"] \arrow[d, "\mu^T_S"'] & TST \arrow[r, "\chi_T"] & STT \arrow[d, "S\mu^T"] \\ TS \arrow[rr, "\chi"'] & & ST \end{tikzcd} \qquad \begin{tikzcd} & S \arrow[dl, "\eta^T_S"'] \arrow[dr, "S\eta^T"] & \\ TS \arrow[rr, "\chi"'] & & ST \end{tikzcd}$$

all commute.

ct: (ST, ν, ξ) is a monad where

$$\nu : \underline{1} \xrightarrow{\eta^S \eta^T} ST$$

$$\xi : STST \xrightarrow{S\lambda_T} SSTT \xrightarrow{\mu^S \mu^T} ST .$$

Meta-lecture 2

Recall from last time:

- Have a clear point you are trying to make.
- Correctness matters.
- Don't try to be interesting.
- Questions from the audience matter.

Remember also: you may present in French, and if you do choose to present in English we will not penalise you for linguistic mistakes, explicitly:

- forgetting an English word,
- getting grammar wrong,
- switching momentarily to French,
- etc

is all <u>perfectly fine</u> and will <u>not</u> be penalised.

Marking criterion: (approximate)

General structure
General relevance
General correctness

and you must have 1 example (at least) and 1 technical component (at least), ie, a proof, a calculation, etc.

Preparing your talk:

You are likely to bite off more than you can chew initially, that is fine. The process of writing a talk normally looks like this:

Idea 1: I will talk about Algebraic Geometry!

Idea 2: I will prove that there is no quintic formula!

Idea 3: I will give an example of the quartic formula.

Idea 4: I will derive the quadratic formula

Idea 5: I will give some examples of quadratics and their zeroes.

⋮ etc

As you do your research and become overwhelmed, just specify further, it's fine, it's not about blowing us away with the extraordinarily abstract concept you learnt, it's about giving a good talk.

On the other hand, too simple is also bad, and it is not on to present what has already been presented in the course.

...ticing your talk:

Time yourself.

Practice physically writing your talk at a black board.

Film yourself if possible.

Watching yourself back will be absolutely excrutiating, but it will <u>significantly</u> improve you.

Remember the conversation about jazz music?

It's the notes they're not playing...

What <u>NOT</u> to put in your talk:

- Elongated opinion pieces about what could be, what should be, and what might be.
- Irrelevancies, remember that succinctness is good for enlightenment. You might be surprised by how tempted you are to put in irrelevant material.
- It is ok not to prove every statement you make, just don't be lazy and don't write something you haven't previously proven.

At this stage you should:

- Know with certainty what your topic is.
- Speak to me about what you have by the end of class.
- Have a rough plan of the structure.