We'd like to compose α with β, which is a natural transformation from functor G to H. The component of β at a is a morphism:

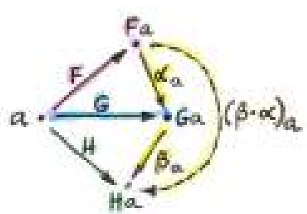
$$\beta_a :: Ga \to Ha$$

These morphisms are composable and their composition is another morphism:

$$\beta_a \circ \alpha_a :: Fa \to Ha$$

We will use this morphism as the component of the natural transformation $\beta \cdot \alpha$ — the composition of two natural transformations β after α:

$$(\beta \cdot \alpha)_a = \beta_a \circ \alpha_a$$



One (long) look at a diagram convinces us that the result of this composition is indeed a natural transformation from F to H:

$$Hf \circ (\beta \cdot \alpha)_a = (\beta \cdot \alpha)_b \circ Ff$$

をūと合成したい。これはファンクタūから☒への自然な変換である。　L1_1D44E における ☒ の構成要素はモルヒズムである。

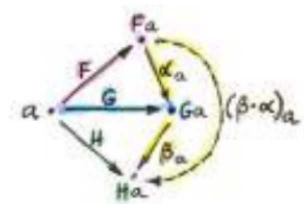
$$\beta_a :: Ga \to Ha$$

これらのモルヒズムは合成可能で、その合成は別のモルヒズムになる。

$$\beta_a \circ \alpha_a :: Fa \to Ha$$

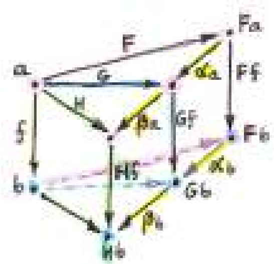
このモルヒズムを自然変換☒・☒ – ☒の後の2つの自然変換☒の合成の成分として使うことにする。

$$(\beta \cdot \alpha)_a = \beta_a \circ \alpha_a$$



図を見てみると、この合成の結果は、確かにFからHへの自然変換であることが納得できる。

$$Hf \circ (\beta \cdot \alpha)_a = (\beta \cdot \alpha)_b \circ Ff$$



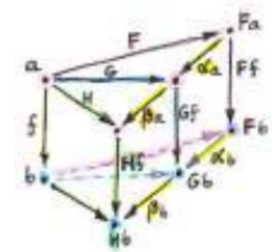
Composition of natural transformations is associative, because their components, which are regular morphisms, are associative with respect to their composition.

Finally, for each functor F there is an identity natural transformation 1_F whose components are the identity morphisms:

$$\mathbf{id}_{Fa} :: Fa \to Fa$$

So, indeed, functors form a category.

A word about notation. Following Saunders Mac Lane I use the dot for the kind of natural transformation composition I have just described. The problem is that there are two ways of composing natural transformations. This one is called the vertical composition, because the functors are usually stacked up vertically in the diagrams that describe it. Vertical composition is important in defining the functor category. I'll explain horizontal composition shortly.



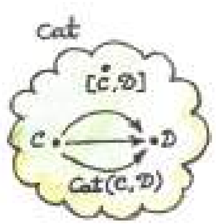
自然変換の合成は、その構成要素である正則モルヒズムが、その合成に関して連想的であるため、連想的である。最後に、各ファンクタFに対して、その構成要素が恒等モルヒズムである恒等自然変換1 ǔが存在します。

$$\mathbf{id}_{Fa} :: Fa \to Fa$$

というわけで、実際、ファンクタはカテゴリを形成する。表記法について一言。Saunders Mac Lane に従って、私は今説明したような自然変換の合成にドットを使うことにしました。問題は、自然変換の合成には2つの方法があることだ。この1つは垂直合成と呼ばれ、それを説明する図ではファンクターが通常垂直に積み重ねられているからである。垂直合成はファンクタカテゴリを定義する上で重要である。水平方向の合成については，簡単に説明しよう．

The functor category between categories **C** and **D** is written as $\mathbf{Fun}(\mathbf{C}, \mathbf{D})$, or $[\mathbf{C}, \mathbf{D}]$, or sometimes as $\mathbf{D}^{\mathbf{C}}$. This last notation suggests that a functor category itself might be considered a function object (an exponential) in some other category. Is this indeed the case?

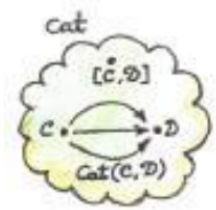
Let's have a look at the hierarchy of abstractions that we've been building so far. We started with a category, which is a collection of objects and morphisms. Categories themselves (or, strictly speaking *small* categories, whose objects form sets) are themselves objects in a higher-level category **Cat**. Morphisms in that category are functors. A Hom-set in **Cat** is a set of functors. For instance $\mathbf{Cat}(\mathbf{C}, \mathbf{D})$ is a set of functors between two categories **C** and **D**.



A functor category $[\mathbf{C}, \mathbf{D}]$ is also a set of functors between two categories (plus natural transformations as morphisms). Its objects are the same as the members of $\mathbf{Cat}(\mathbf{C}, \mathbf{D})$. Moreover, a functor category, being a category, must itself be an object of **Cat** (it so happens that the functor category between two small categories is itself small). We have a relationship between a Hom-set in a category and an object in the same category. The situation is exactly like the exponential object that we've seen in the last section. Let's see how we can construct the latter in **Cat**.

As you may remember, in order to construct an exponential, we need to first define a product. In **Cat**, this turns out to be relatively easy,

カテゴリ ⊠ と ⊠ の間のファンクタ・カテゴリは ⊠，⊠，または ⊠，と書かれることがある。この最後の表記は、ファンクター・カテゴリーそのものが、他のカテゴリーにおける関数オブジェクト（指数関数）と考えられることを示唆している。本当にそうなのだろうか？これまで構築してきた抽象化の階層を見てみよう．まず、オブジェクトとモルヒズムの集合体であるカテゴリーから始めた。カテゴリ自体（厳密には、オブジェクトが集合を形成する小カテゴリ）は、それ自体が上位カテゴリ ⊠⊠ のオブジェクトである。そのカテゴリのモルヒズムはファンクターである。　のホム集合はファンクタの集合である。たとえば、　（　，⊠)は2つのカテゴリ⊠と⊠の間のファンクタの集合である。



ファンクタカテゴリ[⊠，⊠]は、2つのカテゴリ間のファンクタの集合でもある（さらにモルヒズムとして自然変換もある）。その対象は⊠(⊠，⊠)のメンバと同じである。また、ファンクターカテゴリはカテゴリである以上、それ自体が　⊠のオブジェクトでなければならない（たまたま、2つの小カテゴリ間のファンクターカテゴリはそれ自体が小カテゴリであるのだ）。あるカテゴリーにおけるホムセットと、同じカテゴリーにおけるオブジェクトの関係がある。この状況は、前節で見た指数関数的なオブジェクトと全く同じである。後者を⊠⊠でどのように構成するか見てみましょう。覚えているかもしれませんが、指数を構成するためには、まず、積を定義する必要があります。⊠⊠では、これは比較的簡単であることがわかります。

because small categories are *sets* of objects, and we know how to define Cartesian products of sets. So an object in a product category $\mathbf{C} \times \mathbf{D}$ is just a pair of objects, (c, d), one from $\mathbf{C}$ and one from $\mathbf{D}$. Similarly, a morphism between two such pairs, (c, d) and (c', d'), is a pair of morphisms, (f, g), where $f :: c \to c'$ and $g :: d \to d'$. These pairs of morphisms compose component-wise, and there is always an identity pair that is just a pair of identity morphisms. To make the long story short, **Cat** is a full-blown Cartesian closed category in which there is an exponential object $\mathbf{D}^{\mathbf{C}}$ for any pair of categories. And by "object" in **Cat** I mean a category, so $\mathbf{D}^{\mathbf{C}}$ is a category, which we can identify with the functor category between $\mathbf{C}$ and $\mathbf{D}$.

10.4 2-Categories

With that out of the way, let's have a closer look at **Cat**. By definition, any Hom-set in **Cat** is a set of functors. But, as we have seen, functors between two objects have a richer structure than just a set. They form a category, with natural transformations acting as morphisms. Since functors are considered morphisms in **Cat**, natural transformations are morphisms between morphisms.

This richer structure is an example of a **2**-category, a generalization of a category where, besides objects and morphisms (which might be called 1-morphisms in this context), there are also 2-morphisms, which are morphisms between morphisms.

In the case of **Cat** seen as a **2**-category we have:

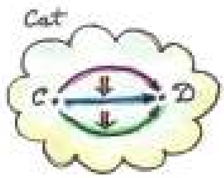
- Objects: (Small) categories
- 1-morphisms: Functors between categories
- 2-morphisms: Natural transformations between functors.

なぜなら、小カテゴリはオブジェクトの集合であり、集合のデカルト積を定義する方法を知っているからです。つまり、積カテゴリ　Lu_1D402 ×　Lu_1D403 のオブジェクトは、　Lu_1D402 のオブジェクトと　Lu_1D403 のオブジェクトのペア、（）だけである。同様に、（☒）と（　Ll_1D4451）という二つの対の間の形態素は、（☒）、☒、☒ という形態素の対で、Ll_1D451 → となる。これらのモルヒズムの対は成分的に合成され、単なる同一性モルヒズムの対が必ず存在する。長い話を短くすると、　は本格的なデカルト閉カテゴリであり、このカテゴリには任意のカテゴリのペアに対して指数関数的なオブジェクト　　が存在することになります。そして、　の「目的語」とはカテゴリーのことなので、☒はカテゴリーであり、☒と☒の間のファンクトルカテゴリーと同定することができる。

10.4 2-Categories

それでは、☒☒をもう少し詳しく見てみましょう。定義によれば、☒の任意のホム集合はファンクタの集合である。しかし、これまで見てきたように、2つのオブジェクトの間のファンクタは単なる集合よりも豊かな構造を持っています。それらはカテゴリーを形成し、自然変換がモルヒズムとして作用する。☒☒ではファンクタは形態素とみなされるので、自然変換は形態素の間の形態素となる。このような豊かな構造は⑷カテゴリーの一例で、オブジェクトとモルヒズム（この文脈では1-モルヒズムと呼ぶ）のほかに、モルヒズム間のモルヒズムである2-モルヒズムも存在するカテゴリーの一般化である。☒☒を　カテゴリとして見た場合、次のようになります。

－ 目的語：（小）カテゴリー － 1-モーフィズム。カテゴリー間のファンクター － 2-モーフィズム。ファンクタ間の自然変換。



Instead of a Hom-set between two categories **C** and **D**, we have a Hom-category — the functor category $\mathbf{D}^{\mathbf{C}}$. We have regular functor composition: a functor F from $\mathbf{D}^{\mathbf{C}}$ composes with a functor G from $\mathbf{E}^{\mathbf{D}}$ to give $G \circ F$ from $\mathbf{E}^{\mathbf{C}}$. But we also have composition inside each Hom-category — vertical composition of natural transformations, or 2-morphisms, between functors.

With two kinds of composition in a **2**-category, the question arises: How do they interact with each other?

Let's pick two functors, or 1-morphisms, in **Cat**:

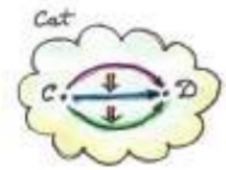
$$F :: \mathbf{C} \to \mathbf{D}$$
$$G :: \mathbf{D} \to \mathbf{E}$$

and their composition:

$$G \circ F :: \mathbf{C} \to \mathbf{E}$$

Suppose we have two natural transformations, α and β, that act, respectively, on functors F and G:

$$\alpha :: F \to F'$$
$$\beta :: G \to G'$$



2つのカテゴリ ⊠ と ⊠ の間のホムセットではなく、ホムカテゴリ - functor category ⊠ がある。⊠からのファンクタ ⊠ は ⊠からのファンクタ ⊠ と合成し、 ⊠ を与える：通常のファンクタ合成がある。しかし、各ホムカテゴリの内部でも、ファンクタ間の自然変換（2-morphism）の垂直合成を行うことができます。1つの⊠カテゴリーに2種類の合成があるとき、疑問が生じる。この2つはどのように相互作用するのでしょうか？⊠⊠で2つのファンクター（1-morphisms）を選んでみましょう。

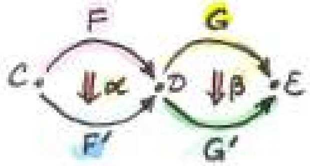
$$F :: \mathbf{C} \to \mathbf{D}$$
$$G :: \mathbf{D} \to \mathbf{E}$$

and their composition:

$$G \circ F :: \mathbf{C} \to \mathbf{E}$$

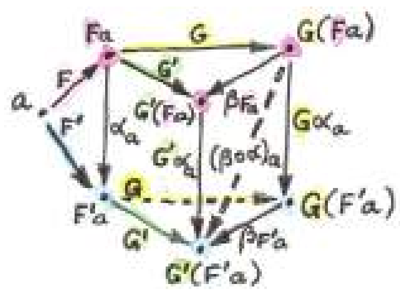
2つの自然変換、⊠と⊠があり、それぞれファンクタ⊠と⊠に作用するとします。

$$\alpha :: F \to F'$$
$$\beta :: G \to G'$$



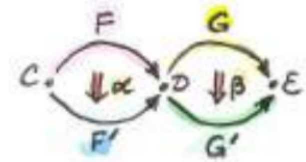
Notice that we cannot apply vertical composition to this pair, because the target of α is different from the source of β. In fact they are members of two different functor categories: $\mathbf{D}^{\mathbf{C}}$ and $\mathbf{E}^{\mathbf{D}}$. We can, however, apply composition to the functors F' and G', because the target of F' is the source of G' — it's the category $\mathbf{D}$. What's the relation between the functors $G' \circ F'$ and $G \circ F$?

Having α and β at our disposal, can we define a natural transformation from $G \circ F$ to $G' \circ F'$? Let me sketch the construction.

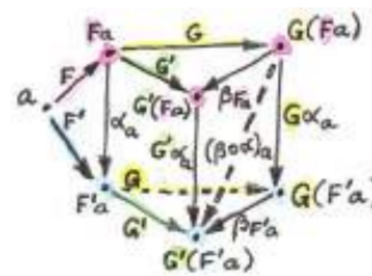


As usual, we start with an object a in $\mathbf{C}$. Its image splits into two objects in $\mathbf{D}$: Fa and $F'a$. There is also a morphism, a component of α, connecting these two objects:

$$\alpha_a :: Fa \to F'a$$



⊠の対象は⊠の元とは異なるので、このペアに垂直合成を適用できないことに 注意してください。実際、これらは⊠と⊠という2つの異なるファンクタカテゴリのメンバである。しかし、ファンクタ ⊠ と ⊠ に合成を適用することはできます。なぜなら、 ⊠ の対象は ⊠ の元であり、それはカテゴリ ⊠ であるためです。ファンクタ ⊠ ′ ⊠ ⊠ の関係はどうなっていますか？⊠とūを手に入れたら、 から への自然変換を定義できるかな？その構成をスケッチしてみよう。



いつものように、⊠の中の物体 L1_1D44E から始める。そのイメージは⊠の2つのオブジェクト、⊠と L1_1D44E に分割される。また、この2つのオブジェクトを結ぶ、 L1_1D6FC の構成要素であるモルヒズムが存在する。

$$\alpha_a :: Fa \to F'a$$

When going from **D** to **E**, these two objects split further into four objects: $G(Fa)$, $G'(Fa)$, $G(F'a)$, $G'(F'a)$. We also have four morphisms forming a square. Two of these morphisms are the components of the natural transformation β:

$$\beta_{Fa} :: G(Fa) \to G'(Fa)$$
$$\beta_{F'a} :: G(F'a) \to G'(F'a)$$

The other two are the images of α_a under the two functors (functors map morphisms):

$$G\alpha_a :: G(Fa) \to G(F'a)$$
$$G'\alpha_a :: G'(Fa) \to G'(F'a)$$

That's a lot of morphisms. Our goal is to find a morphism that goes from $G(Fa)$ to $G'(F'a)$, a candidate for the component of a natural transformation connecting the two functors $G \circ F$ and $G' \circ F'$. In fact there's not one but two paths we can take from $G(Fa)$ to $G'(F'a)$:

$$G'\alpha_a \circ \beta_{Fa}$$
$$\beta_{F'a} \circ G\alpha_a$$

Luckily for us, they are equal, because the square we have formed turns out to be the naturality square for β.

We have just defined a component of a natural transformation from $G \circ F$ to $G' \circ F'$. The proof of naturality for this transformation is pretty straightforward, provided you have enough patience.

We call this natural transformation the *horizontal composition* of α and β:

$$\beta \circ \alpha :: G \circ F \to G' \circ F'$$

⊠から　に行くとき、この2つの対象はさらに4つの対象に分かれる：　Lu_1D43A(⊠　Ll_1D44E),　Lu_1D43A(　),　Lu_1D43A(　Ll_1D43E)；⊠　Ll_1D44E)；．また、正方形を形成するモルヒズムが4つある。これらのモルヒズムのうち2つは自然変換⊠の成分である。

$$\beta_{Fa} :: G(Fa) \to G'(Fa)$$
$$\beta_{F'a} :: G(F'a) \to G'(F'a)$$

他の2つは2つのファンクタのもとでの⊠の像である（ファンクタはモルヒズムを写像する）。

$$G\alpha_a :: G(Fa) \to G(F'a)$$
$$G'\alpha_a :: G'(Fa) \to G'(F'a)$$

モルヒズムがたくさん出てきましたね。我々の目標は、　Lu_1D43A から ⊠（Lu_1D44E）へ行くモルヒズムを見つけることで、2つのファンクタ　と ⊠　Ll_1D44E を結ぶ自然変換の構成要素の候補となるものである。実際、　Lu_1D43A（⊠）から　Lu_1D43A′（⊠）への道は一つではなく、二つあるのです。

$$G'\alpha_a \circ \beta_{Fa}$$
$$\beta_{F'a} \circ G\alpha_a$$

幸運なことに、この2つは等しい。なぜなら、今作った正方形は ⊠ の自然数正方形であることがわかったからである。今、　から　′　⊠への自然変換の成分を定義しました。この変換の自然性の証明は、十分な忍耐力があれば、非常に簡単である。この自然変換を ⊠ と ⊠ の水平合成と呼ぶことにする。

$$\beta \circ \alpha :: G \circ F \to G' \circ F'$$

Again, following Mac Lane I use the small circle for horizontal composition, although you may also encounter star in its place.

Here's a categorical rule of thumb: Every time you have composition, you should look for a category. We have vertical composition of natural transformations, and it's part of the functor category. But what about the horizontal composition? What category does that live in?

The way to figure this out is to look at **Cat** sideways. Look at natural transformations not as arrows between functors but as arrows between categories. A natural transformation sits between two categories, the ones that are connected by the functors it transforms. We can think of it as connecting these two categories.

Let's focus on two objects of **Cat** — categories **C** and **D**. There is a set of natural transformations that go between functors that connect **C** to **D**. These natural transformations are our new arrows from **C** to **D**. By the same token, there are natural transformations going between functors that connect **D** to **E**, which we can treat as new arrows going from **D** to **E**. Horizontal composition is the composition of these arrows.

We also have an identity arrow going from **C** to **C**. It's the identity natural transformation that maps the identity functor on **C** to itself. Notice that the identity for horizontal composition is also the identity for vertical composition, but not vice versa.

Finally, the two compositions satisfy the interchange law:

$$(\beta' \cdot \alpha') \circ (\beta \cdot \alpha) = (\beta' \circ \beta) \cdot (\alpha' \circ \alpha)$$

ここでも、Mac Laneに従って、水平合成には小さな円を使いますが、その代わりに星を使うこともあります。ここで、経験則のようなものがあります。構図があるたびに、カテゴリーを探す必要があります。自然変換の垂直合成はファンクタ・カテゴリーの一部である。しかし、水平方向の合成はどうだろうか？それはどのようなカテゴリーに属するのだろうか？これを理解する方法は、□□を横向きに見ることである。自然変換をファンクタ間の矢印としてではなく、カテゴリ間の矢印として見てください。自然変換は2つのカテゴリーの間に位置し、そのカテゴリーは変換するファンクターによって接続されています。この2つのカテゴリーをつなぐと考えることができる。

□の2つの対象、カテゴリー　と□に注目しよう。　Lu_1D402 と　Lu_1D403 をつなぐファンクタの間を行き来する自然変換のセットが存在します。これらの自然変換は、　Lu_1D402 から　Lu_1D403 への新しい矢印である。同様に、　Lu_1D403 から　Lu_1D404 に接続するファンクタの間に行く自然変換があり、これは □ から □ に行く新しい矢印として扱うことができる。水平合成はこれらの矢印の合成である。また、□から□に向かう同一性矢印がある。これは　Lu_1D402 上の恒等ファンクタをそれ自身に写す恒等自然変換である。水平合成の恒等式は垂直合成の恒等式でもあるが、その逆はないことに注意。最後に、2つの合成は交換法則を満たす。

$$(\beta' \cdot \alpha') \circ (\beta \cdot \alpha) = (\beta' \circ \beta) \cdot (\alpha' \circ \alpha)$$

I will quote Saunders Mac Lane here: The reader may enjoy writing down the evident diagrams needed to prove this fact.

There is one more piece of notation that might come in handy in the future. In this new sideways interpretation of **Cat** there are two ways of getting from object to object: using a functor or using a natural transformation. We can, however, re-interpret the functor arrow as a special kind of natural transformation: the identity natural transformation acting on this functor. So you'll often see this notation:

$$F \circ \alpha$$

where F is a functor from $\mathbf{D}$ to $\mathbf{E}$, and α is a natural transformation between two functors going from $\mathbf{C}$ to $\mathbf{D}$. Since you can't compose a functor with a natural transformation, this is interpreted as a horizontal composition of the identity natural transformation 1_F after α.

Similarly:

$$\alpha \circ F$$

is a horizontal composition of α after 1_F.

10.5 Conclusion

This concludes the first part of the book. We've learned the basic vocabulary of category theory. You may think of objects and categories as nouns; and morphisms, functors, and natural transformations as verbs. Morphisms connect objects, functors connect categories, natural transformations connect functors.

But we've also seen that, what appears as an action at one level of abstraction, becomes an object at the next level. A set of morphisms turns into a function object. As an object, it can be a source or a target of another morphism. That's the idea behind higher order functions.

ここでSaunders Mac Laneの言葉を引用しておこう。読者はこの事実を証明するのに必要な明白な図を書き留めることを楽しむことができるだろう。もう1つ、将来役に立つかもしれない表記がある。この新しい ☒ Ll_1D41A の横並び解釈では、オブジェクトからオブジェクトへの移動は、ファンクタを使う方法と自然変換を使う方法の2つがある。しかし、ファンクタの矢印を特殊な自然変換、つまりこのファンクタに作用する恒等自然変換として再解釈することができます。つまり、このような表記をよく見かけます。

$$F \circ \alpha$$

ここで、 Lu_1D439 は ☒ から ☒ へのファンクタで、 Lu_1D402 から ☒ への2つのファンクタ間の自然変換を意味します。ファンクタと自然変換を合成することはできないので、これは Ll_1D6FC の後に ID 自然変換 1 ☒ の水平合成として解釈される。同様に

$$\alpha \circ F$$

は1 ☒の後の☒の水平合成です。

10.5 Conclusion

これで第一部は終わりです。カテゴリー理論の基本的な語彙を学びました。オブジェクトとカテゴリは名詞、モルヒズム、ファンクタ、自然変換は動詞と考えるとよいでしょう。モルヒズムは対象を結び、ファンクタはカテゴリを結び、自然変換はファンクタを結びます。しかし、ある抽象的なレベルでは行為に見えるものが、次のレベルでは物体になることも見てきた。モルヒズムの集合は、関数オブジェクトに変わる。オブジェクトとして、それは別のモルヒズムのソースにもターゲットにもなりうる。それが高階関数の考え方である。

A functor maps objects to objects, so we can use it as a type constructor, or a parametric type. A functor also maps morphisms, so it is a higher order function — `fmap`. There are some simple functors, like `Const`, product, and coproduct, that can be used to generate a large variety of algebraic data types. Function types are also functorial, both covariant and contravariant, and can be used to extend algebraic data types.

Functors may be looked upon as objects in the functor category. As such, they become sources and targets of morphisms: natural transformations. A natural transformation is a special type of polymorphic function.

10.6 Challenges

1. Define a natural transformation from the `Maybe` functor to the list functor. Prove the naturality condition for it.
2. Define at least two different natural transformations between `Reader ()` and the list functor. How many different lists of `()` are there?
3. Continue the previous exercise with `Reader Bool` and `Maybe`.
4. Show that horizontal composition of natural transformation satisfies the naturality condition (hint: use components). It's a good exercise in diagram chasing.
5. Write a short essay about how you may enjoy writing down the evident diagrams needed to prove the interchange law.
6. Create a few test cases for the opposite naturality condition of transformations between different `Op` functors. Here's one choice:

ファンクタはオブジェクトをオブジェクトに写すので、型コンストラクタやパラメトリック型として使うことができます。また，ファンクタはモルヒズムを写すので，高階関数 – fmap – にもなります．Const，product，coproduct などの簡単なファンクタがあり，これらを用いてさまざまな代数的データ型を生成することができます．また，関数型も共変量，共変量ともにファンクショナルであり，代数的データ型の拡張に利用することができる．ファンクタは、ファンクタカテゴリのオブジェクトと見なすことができる。そのため、自然変換というモルヒズムのソースやターゲットになる。自然変換は多相関数の特殊な型である．

10.6 Challenges

1．Maybeファンクタからlistファンクタへの自然変換を定義しなさい．その自然変換の自然性条件を証明しなさい。2．Reader () と list ファンクタの間に少なくとも2つの異なる自然変換を定義しなさい。()のリストは何種類ありますか？3．4．自然変換の水平合成が自然性条件を満たすことを示せ（ヒント：componentsを使用）。これは、ダイアグラムを追いかける良い練習になります。5．交換法則を証明するために必要な明白な図を書き留めることが楽しいかもしれない，という短いエッセイを書きましょう．6．6．異なるOpファンクタ間の変換の反対の自然性条件について、いくつかのテストケースを作成しなさい。ここに一つの選択肢があります。

```haskell
op :: Op Bool Int
op = Op (\x -> x > 0)
```

and

```haskell
f :: String -> Int
f x = read x
```



```haskell
op :: Op Bool Int
op = Op (\x -> x > 0)
```

and

```haskell
f :: String -> Int
f x = read x
```

178