

Introduction to Category Theory/Categories

from:

en.wikiversity.org/wiki/Introduction_to_Category_Theory/Categories

slightly modified by Armando B. Matos

November 7, 2013

Contents

1	Categories	2
1.1	Definition	2
1.2	Discussion	2
1.3	Examples	3
1.3.1	Sets	3
1.3.2	Monoids	3
1.3.3	All monoids	3
1.3.4	Posets	4
1.3.5	All posets	4
1.4	Free Categories	4
1.4.1	Directed graphs	4
1.4.2	Free category on a (directed) graph	4
1.5	Foundational Issues	5
1.6	“Spinoff” Categories	6
1.6.1	The Opposite Category	6
1.6.2	Pair Categories	7
1.6.3	Slice Categories	7
2	Morphisms	8
2.1	Monomorphism	8
2.2	Epimorphism	9
2.2.1	Example: Non-surjective epimorphism	9
2.3	Isomorphism	10
2.3.1	Example: morphisms in posets	11
2.3.2	Exercise: Duals of isomorphisms	11
3	Initial and terminal objects	11
3.1	Examples	11
3.2	Discussion	12
4	Exercises	12

What is this? A Wikipedia paper, by an author designated by “AA”, that introduces important Category Theory (CT) concepts and discusses some interesting points. For students of CT, this revision may be useful.

I have edited and illustrated the the Wikipedia entry. All possible errors and omissions are of course of my entire responsibility.

Armando B. Matos.

1 Categories

1.1 Definition

A category \mathcal{C} is a collection of *objects* $A, B, C \dots$ and *arrows* $f, g, h \dots$ between the objects with following properties:

- Two arrows can be *composed*, if the first arrow ends where the second arrow begins. If f is an arrow from object A to object B and g is an arrow from object B to object C then their composition $g \circ f$, also denoted by $f; g$, is an arrow from A to C .
- Composition of arrows is associative, the order of composition does not matter. $(h \circ g) \circ f = h \circ (g \circ f)$.
- For every object A there is a special arrow $1_A : A \rightarrow A$ called the *identity* arrow of A . For every arrow $f : B \rightarrow A$ and $g : A \rightarrow C$, the identity arrow satisfies $1_A \circ f = f$ and $g \circ 1_A = g$.

Arrows are often called *morphisms*. The collection of all arrows from A to B is written $\text{hom}_{\mathcal{C}}(A, B)$.

1.2 Discussion

These principles are just those of the “algebra of function composition”. So we are making a shift from thinking about functions in terms of what they do to the elements of sets, to thinking about (vaguely) function-like entities that are described in terms of their *external behavior in terms of composition*. But we see from the upcoming list of examples that arrows are only function-like in certain respects; they definitely *do not have to actually be functions*.

From the definition, some basic results follow – in the same way they do for monoids and groups:

- A sequence of arrows a_1, \dots, a_n composes to the same thing no matter how it is parenthesized, as long as the order is kept. e.g. $a \circ (b \circ (c \circ d)) = ((a \circ b) \circ c) \circ d = (a \circ b) \circ (c \circ d)$, etc.
- Identities are unique. That is, if we have some $h : A \rightarrow A$ such that $h \circ f = f$ for all $f : B \rightarrow A$ and $g \circ h = g$ for all $g : A \rightarrow C$, then $h = 1_A$.

The first of these is usually proved by induction in graduate level introductions to abstract algebra (such as Serge Lang’s “Algebra”), but the proof is not very

illuminating (at least to beginners). The proof of the second is however a mini-classic, which you would have already encountered if you have done any group theory:

$$h = h \circ 1_A = 1_A$$

It is the two-sidedness of the property characterizing the identity that makes this work.

1.3 Examples

1.3.1 Sets

Sets is a category whose objects are sets and whose arrows are functions.

There is however a technical point to manage. A function may be presented as a kind of relation between two sets: its domain, which it is “from”, and its codomain, which it is “to”. This is what is needed for category theory, but in the most commonly encountered set-theoretical definition, a function is just a set of ordered pairs meeting the condition that nothing appears as first member of two different pairs – different pairs being those with either different first or second members, by the set-theoretical principle of extensionality. The set of second members is called the *range*, and a function is then “to” *any set that contains the range* as a subset. This would *not work* out for category theory, since we need each arrow to have only one domain and only one codomain.

So to make a set-theoretic function serve as a category arrow in *Sets*, we need to bundle it with with some superset of its range, which then constitutes is *codomain*, so it is an ordered pair ⟨set-theoretic function, codomain⟩. But we do not have to add a domain, since, unlike the range, the domain is determined intrinsically by the set of ordered pairs.

So, having taken all that on board and proved to ourselves that *Sets* is a category, we get for free the results from the Discussion section above that identities are unique.

1.3.2 Monoids

Every monoid is a category with only one object. Category is a generalization of monoid. The section on Free Categories just below expands on this point.

1.3.3 All monoids

The category of monoids *Mon* is a category, whose objects are monoids and whose arrows are monoid homomorphisms.

This represents a very large and important class of examples: for pretty much any of the kinds of algebraic systems you might study in an Abstract Algebra course, there will be a category with instances of the kind of system in question as objects, and homomorphisms (however defined for that particular kind of system) as arrows.

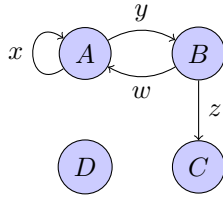


Figure 1: A simple directed graph.

1.3.4 Posets

A partially ordered set or a poset is a category, whose objects are elements of the poset and whose arrows are “less than or equal” relations.

In greater detail, a preorder is a category in which there is at most one arrow from any object A , to any object B . So the relation “there is an arrow from A , to B ”, will satisfy the axioms of reflexivity and transitivity, but not necessarily either symmetry or antisymmetry. We can impose antisymmetry by requiring at most one arrow connecting A and B , regardless of direction. This example provides another instance where arrows are not functions. The Hasse diagram of a poset is a directed graph whose corresponding free category is that set with its partial order.

1.3.5 All posets

The category of posets \mathcal{Pos} is a category, whose objects are posets and whose arrows are order preserving functions. If A and B are posets, then function $f : A \rightarrow B$ is order preserving or monotone, if for every elements x and y in A , $f(x) \leq f(y)$ if $x \leq y$. This is another example where algebraic systems of some type are the objects, and their homomorphisms the arrows.

1.4 Free Categories

1.4.1 Directed graphs

Consider Figure 1. Here we have objects (A, B, C, D) and arrows (w, x, y, z) , the basic furniture of a category, but no category because there are no rules. In particular, the only arrows present are the ones depicted; for instance, there is no identity arrow on B and no arrow $x \circ y$.

A directed graph is a collection of objects and arrows without any rules of composition or identity arrows. We say that a graph is *small* if the collections are sets. A small directed graph can be described as a set O of objects and a set A of arrows, and two functions $\text{dom} : A \rightarrow O$ and $\text{cod} : A \rightarrow O$ that associate domain and codomain to each arrow. Conversely, given any two disjoint sets A and O , and any two functions from A to O , we get a small directed graph.

1.4.2 Free category on a (directed) graph

An obvious way to get a category out of a graph is to let category arrows be (finite) sequences of graph arrows, with the empty sequence $\langle \rangle$ serving as the identity for

each node, and composition of sequences defined in the “obvious” way. Alternatively, we could consider the transitive closure of the relation defined by the graph.

Well, that is the basic idea, but there are a few annoying technicalities. First, and trivially, we have to decide whether to write our sequences in the order of traversal of a path – intuitively natural –, or in the standard order of arrow (and function) composition – counter-intuitive but supported by tradition. We will try to uphold rationality against tradition for a few minutes and write our paths in the order of traversal. So, technically an arrow f in a category freely generated by graph is a sequence $\langle x_1, x_2, \dots, x_m \rangle$ of arrows x_i in graph such that $\text{cod}(x_i) = \text{dom}(x_{i+1})$ for all indexes i . The word “freely” means that two sequences $\langle x_1, x_2, \dots, x_m \rangle$ and $\langle y_1, y_2, \dots, y_n \rangle$ represent the same arrow if and only if they are the same sequence, that is

$$m = n \text{ and } x_1 = y_1, x_2 = y_2, \dots, x_m = y_n$$

Composition of arrows is defined as: if $f = \langle x_1, \dots, x_m \rangle$, $g = \langle y_1, \dots, y_n \rangle$, and $\text{cod}(x_m) = \text{dom}(y_1)$, then:

$$g \circ f = \langle x_1, \dots, x_m, y_1, \dots, y_n \rangle$$

Ugh. Well, eventually stuff like this gets easier to write for yourself than to read. Note the confusing order switch between the structure of the composed sequence and the arrow-composition notation. Associativity of course follows from the associativity of composition of sequences.

The other technical point involves the identities and the requirements for something to be an arrow. **Can you figure out what it is and suggest a solution?**

Hint/Solution. The problem is that arrows are supposed to have unique domains and codomains, so the empty sequence technically can not serve as the identity arrow for every node in the graph (= object in the free category). A reasonable solution is to use the nodes themselves as their own identity arrows, and engineer this by brute force into the definition of composition. \square

We say that a category is *free*, if it is freely generated by some directed graph.

Free categories are of considerable importance for applying category theory to logic, e.g. [Lambek and Scott (1986), “Introduction to Higher Order Categorical Logic”]. They also provide a fine illustration of why arrows do not have to be functions. And they also illustrate the point above that a category is a generalization of a monoid, since the category arrows are like monoid elements that can not always be composed. Monoids in general have one feature that free categories can lack, in that different compositions of arrows can result in the same monoid element. This can happen with categories too, but not with free ones.

1.5 Foundational Issues

There are some tricky foundational points that arise in the definition of category. The word “collection” was used where one might have expected “set”, because in many important examples of categories, the collections of objects and arrows are “too big” to be sets, but instead must be proper classes. The most obvious case is our first example, the category *Sets*, where Russell’s Paradox tells us that there can not be any such thing as a set of all sets, so the collection of objects of *Sets*

can not be a set.

If the collections of objects and arrows of a category are sets, it is called a *small category*, otherwise, a *large category*. So *Sets* is large, and any monoid is small. What about a poset, or *Mon*? Most of the large categories that people are interested in, such as *Sets*, have the property that for any objects A, B , $\text{hom}_{\mathcal{C}}(A, B)$ is a set. Such a category is called *locally small*.

Category theorists appear on the whole to not be very interested in fussing about foundations, so if you can remember what “small”, “large”, and “locally small” mean, you almost certainly know enough.

1.6 “Spinoff” Categories

All algebraic systems provide various further ones of the same kind; for groups, for example, we might have some interesting subgroups and quotient groups, and will always have lots of product groups. Categories seem to participate in the process with an unusual degree of exuberance. For each of the “spinoff categories” below (not a standard term, but it seems quite appropriate to this author (AA)), it is a good basic exercise to prove that it really is a category. The proof is always a routine demonstration that:

- The putative arrow-composition actually produces an arrow in the category.
- Composition is associative.
- Each object has an identity arrow

There are plenty more of these things than we illustrate here, but is it necessary to learn them all at once?

1.6.1 The Opposite Category

For any category \mathcal{C} , we also have the *opposite category* \mathcal{C}^{op} , formed by retaining the objects of \mathcal{C} without change, but swapping the domain and codomain of each arrow. Composition is defined by the equation $f^{\text{op}} \circ g^{\text{op}} = (g \circ f)^{\text{op}}$. Yet another way of making the point that arrows do not have to be functions. Here is an outline of the proof that \mathcal{C}^{op} really is a category. This is also often called the *dual category*.

Hint/Solution. Outline of proof that \mathcal{C}^{op} is a category:

- The result of composition belongs to the opposite category by definition.
- Composition is associative via a chain of identities involving the definition of opposite-composition and the associativity of the original category.
- The identities of the original category serve as identities for the opposite category.

□

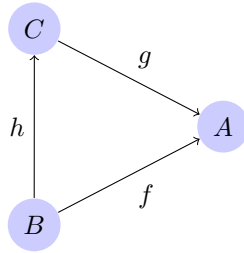


Figure 2: Arrow h in slice category.

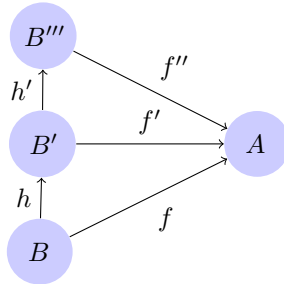


Figure 3: Composition of arrows in slice category.

1.6.2 Pair Categories

For any categories \mathcal{C} , \mathcal{D} , we can form the *pair category* $(\mathcal{C}, \mathcal{D})$ whose objects are pairs (C, D) of objects from \mathcal{C} , \mathcal{D} , respectively, and whose arrows are pairs (f, g) , of arrows of \mathcal{C} , \mathcal{D} , respectively, such that if $f : C \rightarrow C'$, $g : D \rightarrow D'$, then

$$(f, g) : (C, D) \rightarrow (C', D')$$

Composition is defined as:

$$(f', g') \circ (f, g) = (f' \circ f, g' \circ g)$$

This is basically a direct product construction, and generalizes straightforwardly to n -tuples.

1.6.3 Slice Categories

For any object A of a category \mathcal{C} , the *slice category of objects over A* , denoted by \mathcal{C}/A , has as *objects* those *arrows* of \mathcal{C} that have A as their codomain. The arrows are a bit harder to describe. Suppose that $f : B \rightarrow A$ and $g : C \rightarrow A$ are objects of \mathcal{C}/A . Then $h : B \rightarrow C$ of \mathcal{C} is also an arrow of \mathcal{C}/A , from f to g , if $g \circ h = f$. This is illustrated in the *commutative diagram* of Figure 2, which is a graph representing some objects and arrows in the category, and there is a convention that any two paths between the same node are the same arrow (typically derived by a different composition of other arrows).

Commutative diagrams can be formalized as a way to represent facts about categories, e.g. [Barr and Wells 1999:91-94], but we will just take them as suggestive

intuitive representations for equations, which will be the “official” format for imposing identities on different compositions of arrows.

We now need to define composition for the arrows; the basic idea is that the composite of two slice-category arrows is the slice-category correspondent of the two original arrows. We need to do a bit of work to show that this works, as expressed in the diagram of Figure 3. Here h is an arrow $f \rightarrow f'$, and h' , one $f' \rightarrow f''$. For $h' \circ h$ to be an arrow from f to f'' , it must be the case that $f'' \circ h' \circ h = f$, which you can verify as an exercise. Associativity and identities then follow immediately from these properties of the original arrows of \mathcal{C} .

A (perhaps excessively) fine point is that, although people standardly speak of the arrows of \mathcal{C}/A as being certain of the arrows of \mathcal{C} , this can not technically be quite right, since arrows have unique domains and codomains, and these are different for the arrows of the two categories. What we really want is for each suitable arrow of \mathcal{C} to induce a (different) corresponding arrow of \mathcal{C}/A . A construction that would suffice would be to have the arrows of \mathcal{C}/A be triples of the form (original arrow, new domain, new codomain).

2 Morphisms

2.1 Monomorphism

“One-to-oneness”, an “internal” property of function, is equivalent to the “external” property of being cancellable on the left

$$g \circ f_1 = g \circ f_2 \text{ if and only if } f_1 = f_2$$

Arrows that are not functions of any kind can not be one-to-one, but they can have the cancellation property, giving us:

Monomorphism, a categorical analogy of injective function

A *monomorphism* (also called a *monic morphism* or a *mono*) is a morphism $f : X \rightarrow Y$ which is left-cancellative in the following sense

$$\text{for all morphisms } g_1, g_2 : Z \rightarrow X, \quad f \circ g_1 = f \circ g_2 \text{ implies } g_1 = g_2$$

The situation described by these equations is often depicted by this diagram:

$$Z \begin{array}{c} \xrightarrow{g_1} \\ \xrightarrow{g_2} \end{array} X \xrightarrow{f} Y$$

Put another way, map $f : X \rightarrow Y$ is monic if and only if the induced map $f_* : \text{hom}(Z, X) \rightarrow \text{hom}(Z, Y)$ is injective for all Z . Here f_* is function such that $f_*(g) = f \circ g$ for all g in $\text{hom}(Z, X)$. If your mind bounces off this statement at first, it is definitely worth reflecting on until it really seems to make sense.

In the category *Sets* the monomorphisms are exactly the injective functions. Many familiar arrows, such as the homomorphisms of various kinds of algebraic systems, are functions with some additional restrictions placed on them – e.g. they must

preserve some algebraic relationships. Such arrows are (almost?) always one-to-one/injective as functions if and only if they are monic as arrows. Eventually we will be able to formulate a more precise and general version of this claim.

Prove the following two easy results about monomorphisms:

- If f and g are monomorphisms, then $g \circ f$ is one too.
- If $g \circ f$ is a monomorphism, then f is one too.

Hint/Solution. Here are the proofs.

- Suppose f, g are monomorphisms, and $g \circ f \circ h = g \circ f \circ h'$. Then $f \circ h = f \circ h'$ (why?) and so $h = h'$ (why), so $f \circ g$ is a monomorphism.
- Suppose $g \circ f$ is a monomorphism, and $f \circ h = f \circ h'$. Then $g \circ f \circ h = g \circ f \circ h'$, so $h = h'$. Therefore, f is a monomorphism.

If you are new to algebraic proofs, make sure you understand the implicit role of associativity in these arguments □

2.2 Epimorphism

Likewise, there is an “external” version of the “internal” property of onto-ness:

Epimorphism is (almost) a categorical analog of surjective function

An *epimorphism* (also called an *epic morphism* or an *epi*) is a morphism $f : X \rightarrow Y$ which is right-cancellative in the following sense:

for all morphisms $g_1, g_2 : Y \rightarrow Z$, $g_1 \circ f = g_2 \circ f$ implies $g_1 = g_2$

$$X \xrightarrow{f} Y \begin{array}{c} \xrightarrow{g_1} \\ \xrightarrow{g_2} \end{array} Z$$

Put another way, a map $f : X \rightarrow Y$ is epic if and only if the induced map $f^* : \text{hom}(Y, Z) \rightarrow \text{hom}(X, Z)$ is injective for all Z . Here f^* is a function such that $f^*(g) = g \circ f$ for all g in $\text{hom}(Y, Z)$.

In the category *Sets* the epimorphisms are exactly the surjective functions. There is also another way to define surjective, which is equivalent to epimorphism in many interesting categories, but certainly not in all categories.

From the discussion of “opposite categories”, we can see that a monomorphism f is an epimorphism in \mathcal{C}^{op} (or, if we want to be ultra-fussy, corresponds to one), and vice-versa. This gives us free proofs for two results about epimorphisms corresponding to the ones about monomorphisms above. We also say that mono- and epi-morphisms are dual concepts.

2.2.1 Example: Non-surjective epimorphism

Consider the monoid homomorphism f from the natural numbers to the integers that takes every number to itself, that is

$$f : \mathbb{N} \rightarrow \mathbb{Z}, \text{ where } f(n) = n \text{ for all integers } n$$

The morphism f is clearly not surjective as a function in the category of sets. We claim it is an epimorphism in the category of monoids.

Let M be any monoid and let g_1, g_2 be any monoid homomorphisms $\mathbb{Z} \rightarrow M$, such that $g_1 \circ f = g_2 \circ f$. This means that g_1 and g_2 agree on all non-negative numbers. We must show that they agree on negative numbers also. We do this by induction.

$$\begin{aligned}
 g_1(-1) &= g_1(-1) \circ 1_M \\
 &= g_1(-1) \circ g_2(0) \\
 &= g_1(-1) \circ g_2(1 + (-1)) \\
 &= g_1(-1) \circ g_2(1) \circ g_2(-1) \\
 &= g_1(-1) \circ g_2(f(1)) \circ g_2(-1) \\
 &= g_1(-1) \circ g_1(f(1)) \circ g_2(-1) \\
 &= g_1(-1 + 1) \circ g_2(-1) \\
 &= 1_M \circ g_2(-1) = g_2(-1)
 \end{aligned}$$

Now suppose we have proved $g_1(-n) = g_2(-n)$. We prove it for $n + 1$ by $g_1(-(n + 1)) = g_1((-n) + (-1)) = g_1(-n) \circ g_1(-1) = g_2(-n) \circ g_2(-1) = g_2((-n) + (-1)) = g_2(-(n + 1))$.

Hint/Solution.

- *Split epimorphism:* every generalized element in codomain is an image of generalized element in domain. This is equivalent to condition that morphism has right-inverse.

This condition is too strong for many algebraic categories. For example monoid homomorphism from \mathbb{Z} to cyclic monoid, $\mathbb{Z}/n\mathbb{Z}$ is not split epi.

- *S-surjective:* every $S \rightarrow \text{cod}(f)$ is an image of $S \rightarrow \text{dom}(f)$. If S is a separator, every $S \rightarrow \text{cod}(f)$ is monic and category has *subobject classifier* this is equivalent to epimorphism [Lawvere and Rosebrugh P4.10].
Monoid homomorphism $\mathbb{N} \rightarrow \mathbb{Z}$, $n \mapsto n$ is not \mathbb{N} -surjective, since $n \mapsto -n$ is not in the image.

□

2.3 Isomorphism

Isomorphism is a categorical analogy of bijection.

An *isomorphism* is a morphism $f : X \rightarrow Y$ that has an inverse morphism $g : Y \rightarrow X$ such that

$$g \circ f = 1_X \text{ and } f \circ g = 1_Y$$

We say that objects X and Y are isomorphic, denoted $X \cong Y$, if there exists an isomorphism between them.

Inverses are unique, that is if h, g are both inverses of f , then $h = g$. **The easy proof is left to reader as an exercise.** This fact lets us speak of the inverse of f , written f^{-1} . Every isomorphism is monomorphism and epimorphism, but the converse is not true.

2.3.1 Example: morphisms in posets

In a partially ordered set morphisms are “less-than-or-equal” relations. For any two objects a, b in a poset, there is at most one morphism from a to b . It follows that every morphism is monic and epic. On the other hand only morphisms with inverses are identity morphisms (proof: anti-symmetry).

2.3.2 Exercise: Duals of isomorphisms

Prove that if f is an isomorphism in \mathcal{C} , then so is its correspondent (its dual) in \mathcal{C}^{op} .

3 Initial and terminal objects

We will be looking at concepts defined by universal properties, which characterize things up to isomorphism, that is, any two things that have the property are isomorphic. In Category Theory, it is relatively unusual to be interested in actual identity: identity up to isomorphism is most often all that is of interest, in accord with the general focus on external behavior rather than internal construction. The simplest universal properties are those being initial and final objects:

An object A in a category \mathcal{C} is an *initial object*, if for every object C in \mathcal{C} , there is exactly one arrow from A to C . Similarly an object A in a category \mathcal{C} is a *terminal object*, if for every object C in \mathcal{C} , there is exactly one arrow from C to A . Some basic facts about these are:

- A initial object of \mathcal{C} is a terminal object of \mathcal{C}^{op} , and vice-versa.
- Initial and terminal objects are unique up to isomorphism. That is if A and B are both initial objects, then there is isomorphism from A to B .

Proof: exercises.

Arrows as elements: if 1 is a terminal object of \mathbf{Sets} , and A is an object of \mathbf{Sets} , then there is a one-to-one correspondence between the conventional set-theoretic members of A and the arrows from 1 to A .

3.1 Examples

- In the category of sets \mathbf{Sets} the empty set is an initial object. For any set X there is only one function from the empty set to set X , the empty function. Any set with only one element, a singleton set, is a terminal object. In \mathbf{Sets} , there is in fact only one initial object, due to the axiom of extensionality. And, for any set X there is only one function from X to a singleton set, a constant function that has the same value on every element of X . There are many different singleton sets, but they are all isomorphic. These examples provide a further illustration of how “internal” properties can be replaced by “external” ones, since the notions of having zero elements or one can be described (for sets) in terms of the arrows from or to the set. Of course one has to do quite a bit of further development in order to describe all of the properties of sets “externally”.

- The monoid with only one element, the identity element, is both initial and terminal object in the category of monoids. An object that is both initial and terminal is called a *zero object*. The zero object is only unique up to isomorphism.
- In a poset, a minimum element, if there is one, is an initial object, and a maximum element, if there is one, is a terminal object. Here the initial and terminal objects are fully unique.

3.2 Discussion

A few further points:

- An initial object in \mathcal{C} is terminal in \mathcal{C}^{op} (proof: trivial).
- Any arrow $f : 1 \rightarrow A$, where 1 is a terminal object, is monic. **Proof: exercise.**
- An arrow $f : 1 \rightarrow A$ is often called an “element” of A . **Can you identify the property of *Sets* that motivates this terminological usage?**

4 Exercises

1. We say that an arrow $f : A \rightarrow B$ has a left inverse if there is an arrow $g : B \rightarrow A$ so that $g \circ f = 1_A$. Suppose that f has a left inverse and show that f is a monomorphism.
2. Show that an epimorphism with a left inverse is an isomorphism.
3. Give an example of a category and an arrow in that category that is not an isomorphism but is both a monomorphism and epimorphism.