Multiversity *Algebra Chapter 0* Reading Group

# Summaries

Chapter I)

Section 1) Explains fundamentals of set theory and basic set operations

Section 2) Explains set relations, set functions and some more advanced set operations

Section 3) Presents categories, and multiple examples of categories. Some are simple, some are advanced (e.g., types of comma categories).

Section 4) Presents monomorphisms and epimorphisms in more detail, taking care to distinguish general morphisms from set functions and their accolytes (inj, surj, etc).

Section 5) Presents more advanced concepts from category theory, mostly some important universal properties, and gives exercises on how to use and prove these, including using comma categories.

Chapter II)

Section 1) Presents the notion of "group" algebraic structures.

# Group Weekly Reports

Week 1 : Today we mostly talked about the first chapter first section’s reading; going over the vocabulary term by term (see the lexicon on the github repo), and going more in depth over certain concepts (particularly relating to set relations). We also saw a bit of a "teaser" of how these notions are used. We did not go over the exercises since not everyone had done them.

Week 2 : Today we continued on discussing the first chapter, it was mostly freeform. We mostly talked about foundations of set theory (mostly stemming from the discussion of exercise 1 on russell’s paradox), why we use function notation the way we do, and about some of the operators over sets themselves (including through some examples from linear algebra and things like the subobject classifier which is seen at the end of section 3).

Week 3 : Today we finished discussing the first chapter. We went over all exercises. We mostly spoke about equivalence relations and partitions. We also spoke about the geometry/topology of quotients of sets by equivalence relations. This was naturally related to exercises 1.2 to 1.7.

Week 4 : We went over monomorphisms and epimorphisms in more depth. We corrected exercises 2.1 to 2.3 (included)

Week 5 : We went in depth over the distinction between isomorphisms and bijections (foreshadowing a bunch of category theory while we were at it) and corrected exercises 2.4 and 2.5.

Week 6 : We went in depth over the notion of section. We corrected exercises 2.6 and 2.7. For the latter exercise, we understood Tristan’s solution by ourselves ! (written by Amric)

Week 7 : We reviewed the notions of algebraic quotient and well-definition. We broached the notion of universal property. We used this to correct exercises 2.8 and 2.9.

Week 8 : We corrected exercises 2.10 and 2.11. We then did some preliminary explanations to present categories and help with the reading of section 3.

Week 9 : We spoke more in depth about category theory, concrete categories, local smallness, algebraic structures (and their vocabulary) and applied category theory.

Week 10 : We reviewed examples 3.2, 3.3, 3.4 and gave a bunch of disambiguation ideas for 3.5. Next week we’ll go over 3.5 and 3.6 in a bit more detail, and start correcting the exercises for this section. We’ll leave 3.7 and above for when we get to their respective exercises

Week 11 : This week we went over examples 3.5 and 3.6 in a lot more detail, made our own "concrete" example (for 3.6). Next week we’ll start correcting exercises, since examples 3.7 and above have their own exercises.

Week 12 : This week we went over exercises 3.1 to 3.5 (included). We also saw a concrete example of duality (matrix/vector transposition). We also spoke a little about things that Tristan has learned from working on section 5 (comma categories and the formalization of diagrams). Next week (or in 2 weeks) we’ll continue correcting the exercises, including those on the various examples of comma categories.

Week 13 : Today we talked mostly about slice and coslice categories, as well as subcategories, and corrected exercises 3.6, 3.7 and 3.8. For next time, an "extra credit" exercise would be to translate the matrix slice category "BONUS" example into a coslice category example (hint: use transposition).

Week 14 : Today was mostly revisions. We again talked about slice and coslice categories. We reviewed the examples and made things more rigorous, as well as corrected some mistakes that had been lurking. We also added Aluffi’s "example 3.8 concrete coslice" on pointed sets as an example of a coslice category.

Week 15 : We finished the coslice category concrete example, and corrected exercises 3.9 and 3.10.

Week 16 : We gave some examples of bislice, bicoslice, fibered bislice and fibered bicoslice categories.

Week 17 : We moved on to section 4 and discussed in depth the distinctions between monomorphisms and injections, as well as between epimorphisms and surjections. We corrected exercise 4.3 along the way.

Week 18 : We corrected exercises 4.1 and 4.2 and went over 4.3 again.

Week 19 : This week, since a couple of members were absent, we did a quick tour of sections one and two for our new member, while working on a summary lexicon for these sections.

Week 20 : [Last] week we did some reviews, and improved the lexicon, mostly concerning section 3.

Week 21 : This week we spoke about universal properties, initial elements and final elements (terminal elements) and how these latter ones are useful to define universal properties.

Week 22 : Last week (july 2nd) we spoke about quotient spaces and how quotients respect a universal property.

Week 23 : This week (july 9th) we went over the universal property of products and the universal property of coproducts in Set with concrete examples. Next week we’ll start correcting the exercises.

Week 24 : This week we corrected exercises 5.1 to 5.4.

Week 25 : This week we corrected exercise 5.5, and reviewed slices, coslices, initial and final objects for in order to solve it.

Week 26 : Today we completed exercise 5.6.

Week 27 : Today we worked on exercise 5.7 but got confused.

Week 28 : Today we kind’a gave up on 5.7, but solved 5.8.

Week 29 : Today we solved 5.9 and 5.10.

Week 30 : Today we solved 5.11 and discussed approaches for 5.12.

Week 31 : Today we solves 5.12, and with it, completed Chapter 1.

Week 32: Talked about groups, groupoids and orders

Week 33: Talked about groups, orders and lattices, and solved exercise II.1.1

Week 34: Solved exercises 1.2, 1.3, 1.4, 1.5 and 1.6 up to (including) order 3

Week 35: Finished exercise 1.6 and talked about homotopy and whatnot. We cut it short since some had left and others were falling asleep.

Week 36: today we corrected 1.7 to 1.11 (included)

Week 37: Today we corrected exercises II.1.12 to II.1.14 (included). We’re thinking of skipping II.1.15 next week, and going straight to reading section II.2

Week 38: Today we corrected exercise II.1.15

Week 39: we talked a bunch about functors and corrected II.2.1

# Lexicon

## Chapter 1

### Section 1

* Set (not a multiset): the set of all sets
* : the empty set, containing no elements;
* : the set of natural numbers (that is, nonnegative integers);
* : the set of integers;
* : the set of rational numbers;
* : the set of real numbers;
* : the set of complex numbers.
* singleton:
* : existential quantifier, "there exists"
* : universal quantifier, "for all"
* inclusion:
* subset:
* cardinal:
* powerset:
* : the union:
* : the intersection:
* $\\$: the difference:
* : the disjoint union:
* : the (Cartesian) product:
* complement of a subset
* relation
* order relation
* equivalence relation
* reflexivity
* symmetry
* antisymmetry
* transitivity
* partition
* : quotient by an equivalence relation

### Section 2

* function
* graph
* (categorical, function) diagram
* identity function
* kernel (of a function)
* image (of a function)
* restriction (of a function to a subset)
* multiset
* composition
* commutative (diagram)
* injection
* surjection
* bijection
* isomorphism
* inverse
* pre-inverse, right-inverse
* post-inverse, left-inverse
* monomorphism
* epimorphism
* natural projection
* natural injection
* canonical decomposition (of a function)

### Section 3

* category
* object
* morphism
* endomorphism
* operation
* discrete category
* small category
* locally small category
* slice category
* coslice category
* comma category (mentioned, undefined)
* pointed set
* category ?? (bislice, bicoslice, fibered bislice, fibered bicoslice)
* dual category

### Section 4

* groupoid: category in which every morphism is invertible. A category of this sort can be viewed as augmented with a unary operation on the morphisms, called inverse by analogy with group theory.
* automorphism

### Section 5

* universal property
* initial object
* final object
* terminal object
* (categorical) product
* (categorical) coproduct
* (categorical) pullback / fibered product
* (categorical) pushout / fibered coproduct
* (set) pullback / fibered product
* (set) pushout / fibered coproduct

## Chapter 1

### Section 1

* group (groupoid with single object / structure that is associative, unitary, and invertible)
* binary operation
* trivial group
* cancellation
* order (of an element of a group ): the smallest amount of times that this element must be repeatedly given to the binary operation before the result becomes , the identity of the group. If this doesn’t happen, then the order is said to be infinite and we write .
* order (of a group ): a synonym for the cardinal of , the (potentially infinite) amount of elements of .

### Section 2

* symmetric group: the group of all permutations of a set , denoted .
* dihedral group: the group of symmetries of a regular polygon, denoted or .
* cyclic group: a group that is generated by a single element and acts like some kind of "clock", denoted .
* generators of a group: a subset of a group for which the closure under the operation of the group is .

# Exercise solutions

## Chapter I)

### Section 1)

#### 1.1)

In a nutshell, Russell’s paradox proves, by contradiction, that certain mathematical collections cannot be sets. It posits the existence of a "set of all sets that don’t contain themselves". Such a set can neither contain itself (since in that case, it would be a "set that does contain itself", and should be excluded); nor can it exclude it itself (since in that case, it would be a "set that doesn’t contain itself", and should be included).

#### 1.2)

Prove that any equivalence relation over a set defines a partition of .

a) has no empty elements: any element in is part of at least one equivalence class, the class containing at least that element itself. Since there is no equivalence class constructed independently from elements, there are no empty equivalence classes.

b) Elements of are disjoint: suppose there is an element that is part of and , two distinct equivalence classes. and . By transivity through , . Therefore, and are the same equivalence class: . Contradiction. Therefore all elements of are disjoint subsets of .

c) The union of all elements of makes up : suppose such that . From the argument made in (a), exists in at least one equivalence class, the class which contains only itself. This is one of ou sets. Contradiction. Therefore,

#### 1.3)

Given a partition on a set , show how to define a relation on such that is the corresponding partition.

The insight here is to build an equivalence relation such that two elements are equivalent if and only if they are part of the same subset of , which is understood as their common equivalence class.

We define such that .

Let us prove that is an equivalence relation.

a) Reflexivity:

b) Symmetry:

c) Transitivity:

Therefore, is indeed an equivalence relation, and is generated uniquely by the partition.

#### 1.4)

How many different equivalence relations may be defined on the set ?

If we start with the 1 element set, we have only one possible partition, one possible equivalence class.

With the 2 element set, there are 2 partitions, and .

With the 3 element set, there is:

* 1 partition of type 1-1-1: .
* 3 partitions of type 2-1: , , and .
* 1 partition of type 3: .

Hence, there are five equivalence classes on the 3 element set.

See the Bell numbers: https://oeis.org/A000110

#### 1.5)

Give an example of a relation that is reflexive and symmetric, but not transitive. What happens if you attempt to use this relation to define a partition on the set?

Let’s imagine a "similarity relation" we can notate with . We can imagine it to work like a looser version of equality (say for example, if an integer is only away, then it counts as similar).

* reflexive: (an element is always "similar" to itself)
* symmetric: ("similarity" goes both ways)
* not transitive: (just because and are similar, that doesn’t mean works, because it is possible for the "similarity gap" to be too large to qualify as "similar". E.g.: .).

If we use this to define a partition on some set : , there is ambiguity as to which element should go into which equivalence class.

This idea deserves further discussion.

In terms of graph theory, if we express a set with an internal relation as a graph, we can represent elements as nodes and relationships as edges. Reflexivity means that every node has a loop (unary, self-edge). Symmetry means that the graph is not directed (since every relationship goes both ways). Transitivity means that every connected subset of nodes is a maximal clique (synonymously, every connected component is a complete subgraph).

In a relation which is reflexive and symmetric, but not transitive, you would have connected components of this graph which are not cliques. For these, there is ambiguity as to how you would group their nodes. Two obvious choices would be either:

* to remove the minimal number of edges to obtain n distinct cliques (thereby gaining the *transitive restriction* of the relation) from a given non-clique; or
* to complete the connected subgraph into a clique (thereby gaining the *transitive closure* of the relation).

#### 1.6)

Define a relation on the set of real numbers, by setting . Prove that this is an equivalence relation, and find a ’compelling’ description for . Do the same for the relation on the plane defined by declaring .

TODO: forgot to prove that it’s an equivalence relation

means that 2 real numbers differ by an integral amount. This means that the equivalence relation algebraically describes the idea that "with this relation, 2 real numbers are the same iff they have the same fractional component (or for negative numbers)". Eg, , since , etc.

To make an algebraic quotient of a set by an equivalence relation, we take the function which maps each element to its corresponding equivalence class, in the set (partition) containing these equivalence class. Intuitively, this is similar to keeping only one representative element per equivalence class. For the example class above, we can keep the representative . There is such an equivalence class for every fractional part possible, that is, one for every number in . The corresponding map is the "real remainder of division modulo 1". This map is well-defined because each real number has only one output for this map, and all real numbers that are equivalent through are mapped to the same value in the output set.

We should also notice that since , this space loops around on itself. Intuitively, if you increase linearly in the input space , it goes back to after in the output space. This output space is thus a circle of perimeter .

Similarly, means that 2 points in the 2D plane are the same iff they differ in each coordinate by an integral amount. This boils down to combining two such loops from the first part of the exercise: one in the direction and one in the direction: what this gives is the small square , which loops to (resp. ) when (resp. ) is reached. This space functions like a small torus, of area .

### Section 2)

#### 2.1)

How many different bijections are there between a set with elements and itself?

Any bijection is a choice of a pairs from 2 sets of the same size, where each element is used only once, and each pair has one element from each set. At first there are choices in each set. We go through each possible input element in order (no choice), for each one, we pick one amongst possibilities for an output.

There are then choice of output left, etc.

Ccl°:

#### 2.2)

Prove that a function has a right-inverse (pre-inverse) iff it is surjective (can use AC).

Let .

##### 2.2.a)

Suppose that has a right-inverse (pre-inverse). We have

Suppose that is not a surjection. This means

Necessarily, is such an , so . Contradiction.

Ccl°:: f is a surjection.

##### 2.2.b)

Suppose that f is a surjection.

We will construct a pre-inverse for .

The insight here is to realize that a surjection divides its input set into a partition, where each 2-by-2 disjoint subset corresponds to , for every in the output set. More formally, each "fiber" (preimage of a singleton) is a disjoint subset of the input set, and the union of fibers is the input set itself. You can see this in the following diagram:

(add diagram) 1234 to ab 1a 2a (fiber from a) 3b 4b (fiber from b) https://tex.stackexchange.com/questions/157450/producing-a-diagram-showing-relations-between-sets https://tex.stackexchange.com/questions/79009/drawing-the-mapping-of-elements-for-sets-in-latex

Using AC, we select a single element from each such fiber. For each , we name the chosen element. We define as . With this, , and so . Thus, has a preinverse.

A summary of this idea: all surjection preinverses are simply a choice of a representative for each fiber of the surjection as the output to the respective singleton.

#### 2.3)

Prove that the inverse of a bijection is a bijection, and that the composition of two bijections is a bijection.

##### 2.3.a)

Using the fact that a function is a bijection iff it has a two-sided inverse (Corollary 2.2) we can see from this defining fact, that is naturally ’s (unique) two-sided inverse, and so is also a bijection.

##### 2.3.b)

Let be , both bijective (hence with inverses in the respective function spaces). Let and . We have:

Therefore and are two-sided inverses of each other, and thus bijections. From this we conclude that the composition of any two bijections is also a bijection.

#### 2.4)

Prove that ‘isomorphism’ is an equivalence relation (on any set of sets).

##### 2.4.a) Problem statement

Let be a set of sets. We define the relation between the elements of as the following:

Let us show that is an equivalence relation.

##### 2.4.b) Reflexivity

For any set , the identity mapping on is a bijection. This means that , ie, is reflexive.

##### 2.4.c) Symmetry

Therefore, is symmetric.

##### 2.4.d) Transitivity

Let be . Suppose that and . This means , both bijections. Let be . is also a bijection since the composition of two bijections is also a bijection (exercise 2.3).

The existence of implies .

Therefore is transitive.

##### 2.4.e) Conclusion

Isomorphism, , is a relation on an arbitrary set (of sets) which is always reflexive, symmetric and transitive. It is thus an equivalence relation.

#### 2.5)

Formulate a notion of epimorphism and prove that epimorphisms and surjections are equivalent.

See "notes" file: section "Proofs of mono/inj and epi/surj equivalence".

#### 2.6)

With notation as in Example 2.4, explain how any function determines a section of .

A section is the preinverse of a surjection. Here, the surjection in question is the projection of onto .

Let .

We now consider the function which maps an input of to its "geometric representation" (its coordinates in the enclosing space , corresponding to a point of the graph ).

We notice that .

Naturally, , therefore, is a pre-inverse (section) of .

This set of relationships can be expressed in the following commutative diagram:

PS: see "On sections and fibers" in the "notes" file for a worked example.

#### 2.7)

Let be any function. Prove that the graph of is isomorphic to .

Using the elements from the previous exercise, we know that is injective from into . This property is inherited to any restriction of the codomain , and corresponding implied restriction of the domain to . In particular, here, and . We now consider . We can see that is injective from being a restriction of an injective function to a smaller codomain. We also know that is surjective, since its domain is its image. Therefore, is a bijection. This means that .

#### 2.8)

Describe as explicitly as you can all terms in the canonical decomposition of the function defined by . (This exercise matches one assigned previously, which one?)

Firstly, elements of are equivalent by this map (they have the same output) if they vary by from each other. This is a reference to the equivalence relation in exercise 1.6. Therefore, we will use in our decomposition. Obviously, the map from , which maps each element of to respective their equivalence class is a surjection (since there’s no empty equivalence class).

Secondly, as mentioned, we have a bijection between and , the circle group of unit complex numbers, namely , where each element of can be understood to correspond to a (class representative) value in the interval .

Finally, we do the canonical injection of into its superset .

#### 2.9)

Show that if and , and further and , then . Conclude that the operation (as described in §1.4) is well-defined up to isomorphism.

We suppose the aforementioned.

Let be a bijection from , and be a bijection from .

We define the following:

This function is a well-defined function, since : every element of the domain has one, and only one, possible image.

Similarly, we define:

Similarly, because , is well-defined.

Let us study . We have:

Hence, . Similarly, . Therefore, , is a bijection, and .

We’ll now do a shift in notation. Let be some arbitrary sets and . Let be such that , , , and . This means , , , and . It also means and . From the above, this implies .

This means that the disjoint union of and is indeed well-defined, up to isomorphism: so long as 2 respective copies of and are made in a way that their intersection is empty, the 2 respective unions of 1 copy each will be isomorphic.

#### 2.10)

Show that if and are finite sets, then .

The number of functions in can be counted in the following way.

For each element of , of which there are , we can pick any element of as the image; a total of choices per choice of . This means , a total of times. Hence, .

#### 2.11)

In view of Exercise 2.10, it is not unreasonable to use to denote the set of functions from an arbitrary set to a set with elements (say ). Prove that there is a bijection between and the power set of .

Simply put, every subset of is built through a series of choices: for each element in , do we keep the element in our subset (output ) or do we remove it (output ) ? It is then easy to see that such a series of choices can easily be encoded as a unique function in . The totality of such series of choices thus corresponds both to the space , and to the powerset , and there is a bijection between the two.

### Section 3)

#### 3.1)

Let be a category. Consider a structure with:

* ;
* for , objects of (hence, objects of ),

Show how to make this into a category.

##### 3.1.a) Composition

First, to make things clearer and more rigorous, let us distinguish composition in as and composition in as . We define as:

We will now show that with verifies the other axioms of a category (namely identity and associativity of composition).

##### 3.1.b) Identity

Since is a category, since has the same objects, and since, by definition, for all object , we have , we can take every as the same identity in . We can verify that this is compatible with :

##### 3.1.c) Associativity

Using associativity in , we have:

Therefore, is associative.

We conclude that is a category.

#### 3.2)

If is a finite set, how large is ?

We know that, in Set, . From a previous exercise, we know that , therefore .

#### 3.3)

Formulate precisely what it means to say that " is an identity with respect to composition" in Example 3.3, and prove this assertion.

Example 3.3 is that of a category over a set with a (reflexive, transitive) relation , where the objects of the category are the elements of , and the homset between two elements and is the singleton if , and otherwise. Composition is given by transitivity of , where . Reflexivity gives the identities ( for any element ).

In this context, to say that " is an identity with respect to composition" means that we can cancel out an element of the form from a composition.

Formally, we have:

proving that is indeed a post-identity, and a pre-identity, in this context.

#### 3.4)

Can we define a category in the style of Example 3.3, using the relation on the set ?

(Description of example 3.3 in the exercise 3.3 just above.)

Naively, saying like in example 3.3 "there is a singleton homset each time we have ", we cannot define such a category, since is not reflexive, and we would thus lack identity morphisms.

However, in a roundabout way, we can define a category over the *negation* of : "there is a singleton homset each time we DO NOT have ". Namely this corresponds to the relation , which is, itself, reflexive, transitive (and antisymmetric), and is a valid instance of the kind of category presented in example 3.3.

In fact, the pair is an instance of what is called a "totally ordered set", which is a more restrictive kind of "partially ordered set" (also called "poset" for short). Consequently, this kind of category is called a "poset category".

#### 3.5)

Explain in what sense Example 3.4 is an instance of the categories considered in Example 3.3.

(Description of example 3.3 in the exercise 3.3 just above.)

Example 3.4 describes a category where the objects are the subsets of a set (equivalently: elements of the powerset of ), and morphisms between two subsets and of are singleton (or empty) homsets based on whether the inclusion is true (or false).

Inclusion of sets, , is also an order relation, this time between the elements of a set of sets (here, ). This means inclusion is reflexive, transitive, and antisymmetric. This makes a poset category, and thus another instance of example 3.3.

#### 3.6)

Define a category by taking , and , the set of matrices with real entries, for all . (I will leave the reader the task to make sense of a matrix with 0 rows or columns.) Use product of matrices to define composition. Does this category ’feel’ familiar ?

The formulation of the exercise is strange. It says to use the product of matrices to define composition, and to have homsets be sets of matrices, but objects of the category are supposed to be integers. I don’t know of any matrix with real entries that maps an integer to an integer in this way.

We thus infer that the meaning of the exercise can be one of two things.

Either we suppose the set of objects could rather be understood as "something isomorphic to ", ie, the collection of real vector spaces with finite bases (ie, ). In which case, this is just the category of real vector spaces with finite basis (and linear maps as morphisms), which is a subcategory of the category real vector spaces (commonly called ). In this context, any morphism starting from is just the injection of the origin into the codomain; and any morphism ending at is the mapping of all elements to the origin.

Otherwise, we understand this as "yes, the objects of the category are integers: this means you should ignore the actual content of the matrices, and instead consider only their effect on the dimensionality of domains and codomains". In this case, this category is a complete directed graph over where each edge corresponds to the change in dimension (from domain to codomain) caused by a given linear map.

#### 3.7)

Define carefully objects and morphisms in Example 3.7, and draw the diagram corresponding to composition.

Example 3.7 (on coslice categories) refers to example 3.5 (on slice categories). Let’s go over slice categories (since example 3.5 asks the reader to "check all [their various properties]").

##### 3.7.1) Slice categories

Slice categories are categories made by singling out an object (say ) in some parent (larger) category (say ), and studying all morphisms into that object. These morphisms become the objects of a new category (ie, for any of , is an object of the slice category, called in this context). In the slice category, morphisms are defined as those morphism in that preserve composition between 2 morphisms into .

Note that there exist pairs of morphisms and between which there is no morphism that exists in the slice category. One such example we can make is in (see notes "On the morphisms of slice and coslice categories" for more details).

3.7.1.a) Identity

A generic identity morphism is expressed diagrammatically in as:

We can see that since in , this is compatible with the definition of a (pre-/right-)unit morphism in . Also, since the only maps post- are maps from , we have as the (post-/left-)unit for every morphism (ie, .

3.7.1.b) Composition

Taking 3 objects of the slice category (, and ), and two morphisms ( mapping to via a -morphism , and mapping to via a -morphism ), we have that and . This is expressed as the following commutative diagram.

Composition of morphisms is then defined as as a mapping from to , such that . This can be understood through the following commutative diagram:

Which commutes, because we have:

Thus, we have a working composition of morphisms.

3.7.1.c) Associativity

We take 4 objects of the slice category (, , and ), and three morphisms ( mapping to , mapping to , and mapping to ). Using composition defined as above, we have

Through associativity in .

##### 3.7.2) Coslice categories

A coslice category is similar, except it takes the morphisms coming *from* a chosen object , rather than those going *to* this object . Below is a commutative diagram in the style of the one of the textbook for slice categories.

We can similarly show that this also defines a category.

3.7.2.a) Identity

A generic identity morphism is expressed diagrammatically in as:

We can see that since in , this is compatible with the definition of a (post-/left-)unit morphism in . Also, since the only maps pre- are maps from , we have as the (pre-/right-)unit for every morphism (ie, .

3.7.2.b) Composition

Taking 3 objects of the slice category (, and ), and two morphisms ( mapping to via a -morphism , and mapping to via a -morphism ), we have that and . This is expressed as the following commutative diagram.

Composition of morphisms is then defined as as a mapping from to , such that . This can be understood through the following commutative diagram:

Which commutes, because we have:

Thus, we have a working composition of morphisms.

3.7.2.c) Associativity

We take 4 objects of the slice category (, , and ), and three morphisms ( mapping to , mapping to , and mapping to ). Using composition defined as above, we have

Through associativity in .

#### 3.8)

A subcategory of a category consists of a collection of objects of , with morphisms for all objects , in , such that identities and compositions in make into a category. A subcategory is *full* if for all , in . Construct a category of *infinite sets* and explain how it may be viewed as a full subcategory of .

To put it less technically, a "subcategory" is just "picking only certain items of a base category , and making sure that things stay closed uneder morphism composition". It is "full" if *all* morphisms between the objects that remain are also conserved.

We can construct a category of infinite sets by taking all the objects of such that , and only homsets between these objects. This is clearly a subcategory of , since it inherits all identity morphisms, composition works the same, and so does associativity; also, restricting the choice of homsets makes it so that the category is closed (you can’t reach a finite set via a homset that went from an infinite to a finite set).

For this category to not be full, there would need to be some homset that loses a morphism, or fully disappears, in the ordeal. However, there is no restriction as to the kind of morphism that is conserved, so any homset that is kept is identical to its original version. Finally, homsets between infinite sets are also infinite sets, so they don’t disappear in this operation.

Consequently defined as such is a full subcategory of .

#### 3.9)

An alternative to the notion of multiset introduced in §2.2 is obtained by considering sets endowed with equivalence relations; equivalent elements are taken to be multiple instances of elements ’of the same kind’. Define a notion of morphism between such enhanced sets, obtaining a category containing (a ’copy’ of) as a full subcategory. (There may be more than one reasonable way to do this! This is intentionally an open-ended exercise.) Which objects in determine ordinary multisets as defined in §2.2, and how? Spell out what a morphism of multisets would be from this point of view. (There are several natural notions of morphisms of multisets. Try to define morphisms in MSet so that the notion you obtain for ordinary multisets captures your intuitive understanding of these objects.) [§2.2, §3.2, 4.5]

Let us recall how multisets were defined in §2.2. Since duplicate elements do not exist in sets, multisets were instead defined as functions from a set to , the set of (nonzero) positive integers. This allows each element in to have a "count", thereby encoding the intuitive notion of multiset. A similar, and equivalent (isomorphic), way of defining it is *via* pairs , which is simpler to think about. We’ll call this category , for "count multiset" (TODO: probably has a conventional and better name, but I don’t know it). As for morphisms in , we can consider that for any multisets and , the homset from to is simply the set functions from to as usual.

We first notice that if we restrict to only the objects for which all elements have a count of , and where morphisms only ever output to in the second coordinate (a subcategory we can call , for example), we get a "copy" of : and are isomorphic in . This is a full subcategory because there are no morphisms that map counts to anything else than if we restrict our objects to this form; so all morphisms between the kept objects are also kept.

Now let us do a similar construction, but based on equivalence classes instead. We know that each equivalence class over a set corresponds uniquely to a partition of that set. By considering only these partitions (these "sets of sets") as objects, we can build a category (for "equivalence multiset"). The "count" corresponds simply to the cardinal of a top-level element in the partition. For example, the top-level elements of would be understood to have counts , and respectively.

As for morphisms in , they simply map each top-level element of the domain multiset (a distinct subset of the original set) to some other top-level elements in the codomain multiset. This has precisely the same effect as mapping pairs of "value and count" as seen in the previous construction.

In this example, any set itself, when "injected" (by a functor) into would just nest all of its elements into singletons. I.e., in would become in . This also shows how restricting to "only objects that are a set of (toplevel) singletons" makes have a "copy" of as a full subcategory (for similar arguments as above).

Yet another example could be something akin to polynomials with integer coefficients on freeform indeterminates of degree 1 (which would be our set elements); raising the operators one rank, a product of freeform variables with integer powers (multiplicities), etc.

#### 3.10)

Since the objects of a category are not (necessarily) sets, it is not clear how to make sense of a notion of ’subobject’ in general. In some situations it does make sense to talk about subobjects, and the subobjects of any given object in are in one-to-one correspondence with the morphisms for a fixed, special object of , called a subobject classifier. Show that has a subobject classifier.

We define the set , aka the binary alphabet or booleans, as the subobject classifier of . For any subset of , there is a unique map , such that (otherwise , of course, as the equivalence and lack of alternatives to as an output imply). The map always fully describes from its relationship with .

#### 3.11)

Draw the relevant diagrams and define composition and identities for the category mentioned in Example 3.9. Do the same for the category mentioned in Example 3.10. [§5.5, 5.12]

For lack of a better term, we will refer to the categories of the form represented by Example 3.9 as "bi-slice categories". The first part of the exercise is thus asking us to define and explain what "bi-coslice categories" (of the form ) are.

Similarly, we will refer to the categories of the form represented by Example 3.10 as "fibered bi-slice categories". The second part of the exercise is thus asking us to define and explain what "fibered bi-coslice categories" (of the form ) are.

We will, of course, attempt to make more formal and pedagogical all definitions broached in the textbook’s examples as well.

##### 3.11.1) Bi-slice categories

3.11.1.a) Objects and morphisms

To make a bi-slice category , we pick 2 objects and of a base category , and consider for all other objects of , all pairs of morphisms . These pairs of morphisms are the objects of the bi-slice category . Morphisms are defined from an object to an object so that we have both and , for some .

A generic object in is of the form:

3.11.1.b) Morphisms

Morphisms are defined between objects as

such that the following diagram commutes

3.11.1.c) Identity

It is clear that identity morphisms exist for all objects, simply by taking , , and , in the diagram above.

3.11.1.d) Composition

Let be 3 objects of , which we will name , and (and define with the respective triplet for ).

Composition of two morphisms and is defined so that the following diagram commutes.

3.11.1.e) Associativity

Associativity follows from associativity of morphisms in , similarly to what was done for slice categories in exercise 3.7 .

##### 3.11.2) Bi-coslice categories

3.11.2.a) Objects and morphisms

To make a bi-coslice category , we similarly pick 2 objects and of our base category , but instead consider, for all other objects of , all pairs of morphisms .

A generic object in is of the form:

3.11.2.b) Morphisms

Morphisms are defined between objects as

such that the following diagram commutes

3.11.2.c) Identity

It is clear that identity morphisms exist for all objects, simply by taking , , and , in the diagram above.

3.11.2.d) Composition

Let be 3 objects of , which we will name , and (and define with the respective triplet for ).

Composition of two morphisms and is defined so that the following diagram commutes.

3.11.2.e) Associativity

Associativity follows from associativity of morphisms in , similarly to what was done for slice categories in exercise 3.7 .

##### 3.11.3) Fibered bi-slice categories

3.11.3.a) Objects

To build a fibered bi-slice category , one takes a base category , as well as a fixed pair of morphisms and in , that point to a common object of . Our basic "fixed construct" from looks like so:

The role of the category is now to study the morphisms into this construct. A generic object from this new category looks like so:

such that the diagram commutes. This means that valid object in are triplets , with and , such that . In a caricatural way, this boils down to studying "the comparison of the different paths one can use to reach , knowing that the last steps are on one hand, , and on the other, ".

3.11.3.b) Morphisms

Morphisms are defined between objects as:

such that the following diagram commutes

3.11.3.c) Identity

Once again, it is clear that identity morphisms exist for all objects, simply by taking , , and , in the diagram above.

3.11.3.d) Composition

Let be 3 objects of , which we will name , and (and define with the respective triplet for ).

Composition of two morphisms and is defined so that the following diagram commutes.

3.11.3.e) Associativity

Associativity follows from associativity of morphisms in , similarly to what was done for slice categories in exercise 3.7 .

##### 3.11.4) Fibered bi-coslice categories

3.11.4.a) Objects

To build a fibered bi-coslice category , one takes a base category , as well as a fixed pair of morphisms and in , that originate from a common object of . Our basic "fixed construct" from looks like so:

The role of the category is now to study the morphisms from this construct. A generic object from this new category looks like so:

such that the diagram commutes. This means that valid object in are triplets , with and , such that . In a caricatural way, this boils down to studying "the comparison of the different paths one can build by starting from , knowing that the choice of first step is on one hand, , and on the other, ".

3.11.4.b) Morphisms

Morphisms are defined between objects as:

such that the following diagram commutes

3.11.4.c) Identity

Once again, it is clear that identity morphisms exist for all objects, simply by taking , , and , in the diagram above.

3.11.4.d) Composition

Let be 3 objects of , which we will name , and (and define with the respective triplet for ).

Composition of two morphisms and is defined so that the following diagram commutes.

3.11.4.e) Associativity

Associativity follows from associativity of morphisms in , similarly to what was done for slice categories in exercise 3.7 .

### Section 4)

#### 4.1)

Composition is defined for *two* morphisms. If more than 2 morphisms are given, one may compose them in several ways, so that every step only consists in composing 2 morphisms. Prove that for any such valid sequence of morphisms, the order of parentheses doesn’t matter.

This boils down to showing that associativity is a global property, that doesn’t just make parentheses meaningless when there are 3 elements and 2 operators between them, but in general elements with operators between them.

Note: A useful way of visualizing this is representing the order of operations as a binary tree, and noticing that applying associativity (forwards or backwards) is just a tree rotation (resp. right or left) at a given node. Then it is easy to show that one can always obtain a "left comb binary tree". Since every choice of parentheses is equal to this left comb choice, and equality is transitive, every choice of parentheses is equal to every other choice.

To be more rigorous, we will proceed by induction.

Hypothesis: = "for a given , for any valid, composable, ordered sequence of morphisms in our base category , any choice of parentheses to compose elements of this sequence 2-by-2, giving a formula , will lead to the same result, which can be seen by always having ".

Initialization: We initialize at ; the validity is immediate as it is precisely the definition of associativity.

Heredity: We suppose the hypothesis true for a given ; let us show that this implies that the hypothesis is true for .

What this means is that, no matter the composable ordered sequence of functions, for a fixed , the order of parentheses does not matter. Note that though is chosen and fixed; the statement is true for EVERY (ordered, composable) sequence of functions. We add a new function to this sequence. By a simple renaming of the functions, we deduce that it doesn’t matter where we insert , so we’ll insert it at the very right to simplify our argument, giving us the sequence .

Here, there are 3 cases. Either:

* is part of the last composition (i.e., it’s not in a semantically necessary parenthethical grouping; it can be made external to all parentheses),
* is part of the first composition (i.e., the first operation is )
* it isn’t either (it’s inside some non-removable parentheses, and needs to be composed earlier on, but not as the first operation).

If is part of the last composition, then by applying the hypothesis to the terms , we immediately find that our new sequence can be made equal to , which is precisely what we wanted for .

If is part of the first composition, we isolate it so that it isn’t anymore. To do so, we apply "backwards" associativity on the grouping of terms in order to obtain , where is the appropriate choice of such that associativity can be applied (with ). This makes it so that our problem is identical to our final case, solved just below.

If is part of neither the first nor last composition, then we consider the innermost composition to be a single element . We now have a sequence of only terms. We apply our hypothesis . This makes the outermost right term, part of the last composition. Unravelling back into two members, we see that we are back at our initial case, with an arbitrary order of parentheses for the terms, and outermost. We already saw that this implied .

Conclusion: since we have initialization and heredity of our hypothesis in all cases, we can conclude by induction that it is true for all .

#### 4.2)

In Example 3.3 we have seen how to construct a category from a set endowed with a relation, provided the latter is reflexive and transitive. For what types of relations is the corresponding category a groupoid (cf. Example 4.6) ?

We remind example 4.6 : a groupoid is a category in which every morphism is an isomorphism. This means that every morphism needs to be 2-way invertible.

In this context, this means that for every morphism , there should be a corresponding inverse morphism . This property is precisely the symmetry of a relation.

This means that all sets with an equivalence relation can be reconstructed into a groupoid.

#### 4.3)

Let , be objects of a category , and a morphism. Prove that if has a pre-inverse, then is an epimorphism. Show that the converse does not hold, by giving an explicit example of a category and an epimorphism without a pre-inverse.

##### 4.3.a)

has a pre-inverse is an epimorphism

Let be a category. Let , having some pre-inverse which we’ll call :

Let be an arbitrary object of , and $\beta', \beta" \in Hom\_{\mathcal{C}} (B, Z)$:

This means that is an epimorphism.

##### 4.3.b)

is an epimorphism $\;\not\!\!\!\Rightarrow$ has a pre-inverse

As was mentioned in the text, "order" categories (poset categories) where there’s only at most one morphism between any two objects makes it so that every morphism is trivially an epimorphism (i.e., since there is at most one morphism "" between any two elements, so you always have $\beta' = \beta" = \leq$, and since $\beta' = \beta"$ is always true, anything implication with it as a necessary condition is also true, and therefore every morphism is true). However, only identities have any kind of inverse (since they are isomorphisms, they are their own inverse); other combinations of elements go one-way, because of antisymmetry.

See also [here](https://ncatlab.org/nlab/show/partial+order) and [here](https://math.stackexchange.com/questions/81123/examples-of-categories-where-epimorphism-does-not-have-a-right-inverse-not-surj).

#### 4.4)

Prove that the composition of two monomorphisms is a monomorphism. Deduce that one can define a subcategory of a category by taking the same objects as in , and defining to be the subset of consisting of monomorphisms, for all objects , . (Cf. Exercise 3.8; of course, in general is not full in .) Do the same for epimorphisms. Can you define a subcategory of by restricting to morphisms that are not monomorphisms?

##### 4.4.a)

Mono

Let be and be monomorphisms. Let us show that is also a monomorphism.

Let be an arbitrary object of , and $\alpha', \alpha" \in Hom\_{\mathcal{A}} (Z, A)$:

This means that the composition of 2 monomorphisms is always an monomorphism. We can thus make a subcategory. Taking all objects, properties, and homsets of , but restricting the homsets only to the monomorphisms, we know that this makes a new category since it is closed under composition, has identities (which are iso, and *a fortiori* mono) and associativity.

##### 4.4.b)

Epi

Let be and be epimorphisms. Let us show that is also a epimorphism.

Let be an arbitrary object of , and $\beta', \beta" \in Hom\_{\mathcal{C}} (C, Z)$:

This means that the composition of 2 epimorphisms is always an epimorphism. We can thus make a subcategory. Taking all objects, properties, and homsets of , but restricting the homsets only to the epimorphisms, we know that this makes a new category since it is closed under composition, has identities (which are iso, and *a fortiori* epi) and associativity.

##### 4.4.c)

Nonmono and nonepi

We could consider the fact that (TODO prove lol) we can’t obtain a monomorphism from the composition of two non-monomorphisms (you need at least one monomorphism in the mix). However, the real problem is identities. Identities are iso, and thus mono. You can’t make a category without identities, so there is no such . the same reasoning applies to .

#### 4.5)

Give a concrete description of monomorphisms and epimorphisms in the category you constructed in Exercise 3.9. (Your answer will depend on the notion of morphism you defined in that exercise!)

We’ll use our construction, where elements of multisets consisted of a pair of the set-element and its count in the multiset.

We recall that in the way we formulated this, morphisms were just simple set functions on "(element, count)" pairs (i.e., returning any other "(element, count)" pair of the codomain). Let be a morphism of multisets . Labelling the elements of the domain as and of the codomain as with , , and any two indexing sets such that and , we can see that and now just look like "normal" sets.

We now simply recycle the notion of injections and surjections. These form our monomorphisms and epimorphisms respectively.

### Section 5)

#### 5.1)

Prove that a final object in a category is initial in the opposite category

Let be a category. Let be the dual category on . Let be a final object in . This means that for every object in , there is a single morphism from to . We will call this morphism (respectively).

We remind how we defined composition in as , respecting:

In this case, we see that . This implies that the homset contains a single morphism, . This means that is initial in .

#### 5.2)

Prove that is the *unique* initial object in **Set**.

First we will prove that it is initial, then that it is unique.

Initiality: we take an arbitrary set in **Set**. We wish to study $Hom\_{\text{\textbf{Set}}}(\emptyset, Z) = Z^\emptyset$. We recall that functions (in category theory) are defined as "applications" / "mappings" are in traditional set theory (i.e., as a relation between sets where every antecedent in the domain has a singleton image in the codomain; the key point being that "no input has no result when passed through the function"). Let be an initial element in **Set**. We write and . We know that . For to be initial, this is true if and only if for all , and so if and only if . We recall that the empty set is the only set with , therefore .

Now this is already enough to prove unicity, but let us spell it out for pedagogy’s sake.

Unicity: We recall that two objects of **Set** are isomorphic if, and only if, there exists a bijection between them. This is equivalent to saying that two sets have the same cardinal. We once again recall that the empty set is the only set with ; there are no bijections relating to the empty set, other than its identity, the unique morphism in $Hom\_{\text{\textbf{Set}}}(\emptyset, \emptyset)$ . Using proposition 5.4 (that terminal objects are unique up-to-isomorphism), we finally deduce that is the unique initial object in **Set**.

NB: the unique function in is always the empty function.

#### 5.3)

Prove that final objects are unique up to isomorphism.

Let us suppose we have a category with two final objects, and .

For every object of there is at least one element in , namely the identity . If is final, then there is a unique morphism , which therefore must be the identity .

We assumed that and are both final in . Since is final, there is a unique morphism in . Since is final, there is a unique morphism in . Consider ; as observed, necessarily the composite since is final. By the same token . Thus and are inverses of each other, proving is an isomorphism. Since there exists an isomorphism between and , .

#### 5.4)

What are initial and final objects in the category of "pointed sets" (Example 3.8)? Are they unique?

We recall that a pointed set is just a regular set with a special, identified point, and that the category of pointed sets **Set\*** is built upon the same objects as **Set**, but where each object in **Set** is multiplied into copies of itself in **Set\*** (one for each choice of special point; this implies that the empty set is not a part of **Set\***, since it has no point). Morphisms in **Set\*** are set functions, but with the restriction of mapping the special point in the domain to the special point in the codomain.

Given this information, we will prove that the initial and final objects in **Set\*** are the singleton sets.

Let be a singleton set in **Set\***. Let be the single element of ; it is necessarily also the special point, as there is no other choice. For any codomain in **Set\***, the condition that "special points map to special points" restricts our choice of function to the unique function , thus, is initial. If had more than one element, there would exist some (non-singletons) for which the other element would allow another degree of freedom (and thus would not be initial).

Similarly, now studying as a domain and as a codomain, we see that that only function from to is (like in **Set**) the function which maps everything (including ’s special point) to . Thus, is final. If had more than one element, there would similarly be many choices for any of cardinal , so long as the special point maps to the special point.

Every singleton pointed set is both initial and final in **Set\*** and is thus a zero object. These are also the only such pointed sets.

#### 5.5)

What are the final objects in the category considered in §5.3?

The category considered in paragraph 5.3 is the coslice category over some set , written . However, what is presented in this paragraph is some extra structure that arises when considering the statement "The quotient is universal with respect to the property of mapping to a set in such a way that equivalent elements have the same image". We thus give some equivalence relation on and study the quotient set in the general coslice category; to do this, we consider the subcategory of where only such that "equivalence is preserved" (i.e., such that ).

With:

* the canonical surjection of onto its quotient ,
* (resp. ) being some arbitrary function from to some arbitrary (resp. ),
* any function (if it exists) such that
* (resp. ) is the (unique!) function such that (resp. )

The following diagram commutes, and summarizes the situation.

Objects in this category (and *a fortiori* ) are denoted as and are obtained from what used to be *morphisms* (regular functions) in **Set**. Morphisms are mappings such that one exists if and only if , and .

Since the textbook also asks whether such a category has initial objects, we will first also answer this and consider all terminal objects.

The initial object of a general coslice category is . This is easily verified by doing , necessarily , in . This implies that, for the domain in and any codomain , there always exists a unique morphism in , corresponding to the (existing and unique) in . We also see that this object satisfies the "equivalence preservation" condition, hence it exists in , and is also the initial object in .

The below are this description first in , followed by the description in .

Now in .

A general coslice category has a final object (or many final objects ) iff has a final object (or many final objects ). In that case, any final object in corresponds to the unique morphism from to (for any final ) in . Let us verify this.

Let be final in , and be the unique morphism . Let be an arbitrary object of . Let be such that . We consider the following diagram:

Since is final in , is unique and always exists. Also, since is unique and always exist, the choice of is irrelevant: this same works for all choices of for a given arbitrary . This proves that exists and is unique for all . Finally, since works for all choices of , it works for those that satisfy the "equivalence preservation" condition, and so does : this means that is indeed a final object in .

#### 5.6)

Consider the category corresponding to endowing (as in Example 3.3) the set of positive integers with the divisibility relation. Thus there is exactly one morphism in this category if and only if divides without remainder; there is no morphism between and otherwise. Show that this category has products and coproducts. What are their ’conventional’ names? [§VII.5.1]

Like example 3.3, this is a case of "category made from an order relation over a set", since divisibility is an order relation (reflexive, antisymmetric, transitive).

Let us remind the definition of categorical products and coproducts. We consider some general category .

An object is the product of two objects and iff there is a unique morphism (resp. ) from to (resp. ), and for every in , and for every pair of morphisms and , there exists a single morphism such that and . This is summarized in the following commutative diagram.

An object is the coproduct of two objects and iff there is a unique morphism (resp. ) from (resp. ) into , and for every in , and for every pair of morphisms and , there exists a single such that and . This is summarized in the following commutative diagram.

We now return to our "divisibility order category". We write its objects as simple integers, and the (if it exists, unique) morphism representing "divisibility of by " as . The conventional name of the product for this category is "greatest common divisor" (or "meet"), and of the coproduct, "least common multiple" (or "join").

The following commutative diagrams represent this fact. Take two arbitrary naturals and . Any number which divides both and also divides their GCD. Likewise, if is a multiple of both and , then it is a multiple of their LCM.

#### 5.7)

Redo Exercise 2.9 ("Show that if and , and further and , then . Conclude that the operation (as described in §1.4) is well-defined up to isomorphism") this time using Proposition 5.4. (the unicity up-to-isomorphism of terminal objects).

TODO, fix, since $\text{\textbf{Set}}^{A,B}$ and $\text{\textbf{Set}}^{A',B'}$ need to both be treated.

This is what we are give by the prompt:

with and since and .

We define $\text{\textbf{Set}}^{A,B}$ as the "bicoslice category of and over **Set**". Objects in this category are pairs of morphisms from and , respectively, into sets . They can be diagrammed as follows.

Morphisms are defined between objects as

such that the following diagram commutes in **Set**

Let us call the following object of $\text{\textbf{Set}}^{A,B}$, where is any choice of valid disjoint union of and :

By definition of a coproduct, we know that in such a configuration, a morphism from this object into any other object of $\text{\textbf{Set}}^{A,B}$ exists and is unique, and so is the on which it is based. This means that is initial in $\text{\textbf{Set}}^{A,B}$.

TODO fix and explain

We observe that , , and are all injections, and thus are post-cancellable, by maps , , and (which are surjections).

and together define a unique map from to .

We set arbitrary and ; we have and . These define a unique map from to , which we’ll write .

#### 5.8)

Show that in every category the products and are isomorphic, if they exist. (Hint: observe that they both satisfy the universal property for the product of and , then use Proposition 5.4.)

Let us first remind the definition of the product of two sets. It is the set made of all pairs of and (ordered sequences of two elements, where the first element in the sequence comes from and the second comes from ) It is the structure such that the following diagram commutes, and is unique.

Now to extend the diagram to consider both and .

We seek to prove that given such a commutative diagram (with a unique and ), which we will call , then we have .

We define as the "bislice category of and over ". Objects in this category are pairs of morphisms from sets , into and , respectively. They can be diagrammed as follows.

Morphisms are defined between objects as

such that the following diagram commutes in :

We now define the following objects:

and

Using the diagram defined above, and the definition of a product (ie, that the maps from any to it in the appropriate configuration are unique). We deduce that both and are final objects in . Finally, using Proposition 5.4, i.e., that final objects in a category are unique up-to-isomorphism, we conclude that .

#### 5.9)

Let be a category with products. Find a reasonable candidate for the universal property that the product of three objects of ought to satisfy, and prove that both and satisfy this universal property. Deduce that and are necessarily isomorphic.

Given 3 objects of . We propose the following universal property that should respect: for all objects of , and triplet of maps , , and from to , and respectively, there exists a unique triplet-map is unique such that the following diagram commutes.

We will now show that both and satisfy this universal property.

##### 5.9.a)

We start with

and

From both of these universal products, we deduce the following commutative diagram.

##### 5.9.b)

We start with

and

From both of these universal products we deduce the following commutative diagram:

##### 5.9.c)

We now consider the as the "trislice category of , and over ". Objects in this category are of the form:

Morphisms are defined between objects as:

such that there exists a making the following diagram commute in :

We now define the following object:

and using the two universal properties we have shown above, we know that necessarily and are terminal objects in . This is because they are final objects in respective bislice categories, given that and are both unique; and we can compose , etc, to get immediate projections, making them final in the trislice category as well. To be precise:

are both final objects in , as is shown by the following commutative diagram, which is composed from the diagrams above:

We use Proposition 5.4 to deduce that and , as terminal objects in , are necessarily isomorphic. Therefore, given this "associativity up-to-isomorphism" of a triple product, it is legitimate to call (with no parentheses) "the" (unique, up-to-isomorphism) final object in .

#### 5.10)

Push the envelope a little further still, and deﬁne products and coproducts for families (i.e., indexed sets) of objects of a category. Do these exist in **Set**? It is common to denote the product ( times) by .

Let be an (ordered) family of objects in some category .

##### 5.10.a)

A product of the elements of is defined as an object of together with a family of morphisms such that for any object and family of morphisms , there exists a unique morphism such that the following diagram commutes for all :

These exist in **Set** for all finite families of sets, however, for infinite families of sets, their existence is conditional on the (famous) axiom of choice (it is actually precisely the point of the axiom of choice).

##### 5.10.b)

A coproduct of the elements of is defined as an object of together with a family of morphisms such that for any object and family of morphisms , there exists a unique morphism such that the following diagram commutes for all :

These exist in **Set** for all families of sets, however, for infinite families of sets, their existence is conditional on the axiom of choice.

#### 5.11)

Let , resp. , be sets, endowed with equivalence relations , resp. . Define a relation on by setting (This is immediately seen to be an equivalence relation.)

* Use the universal property for quotients (§5.3) to establish that there are functions , and ;
* prove that , with these two functions, satisfies the universal property for the product of and ;
* conclude (without further work) that .

##### 5.11.a)

First, we remind the universal property for quotients. Given an equivalence relation over a set , there is a single map such that, for any map verifying whenever (i.e., the morphism is "well-defined" for the equivalence relation ), there exists a unique map such that . This is summarized in the following commutative diagram:

We now apply the universal property of products to obtain the following commutative diagram:

We then consider a well-defined map (i.e., whenever ). We apply the universal property of quotients to the relation defined over . We define the map as . We define the map as . We remind that such an is unique, and that .

We do a similar construction for with and with . We can do so since these maps are also well-defined. Indeed, we have:

$$\begin{array}{l}
\left\{
\begin{array}{l}
(a\_1, b\_1) \sim (a\_2, b\_2) \\
(a\_1, b\_1) \sim (a\_2, b\_2) \Rightarrow f(a\_1, b\_1) = f(a\_2, b\_2) \\
f = f\_A \circ p\_A \\
f = f\_B \circ p\_B
\end{array}
\right.
\cr \Rightarrow \\
\left\{
\begin{array}{l}
a\_1 \sim\_A a\_2 \text{ and } b\_1 \sim\_B b\_2 \\
f\_A(p\_A(a\_1, b\_1)) = f\_A(p\_A(a\_2, b\_2)) \\
f\_B(p\_B(a\_1, b\_1)) = f\_B(p\_B(a\_2, b\_2))
\end{array}
\right.
\cr \Rightarrow \\
\left\{
\begin{array}{l}
f\_A(a\_1) = f\_A(a\_2) \\
f\_B(b\_1) = f\_B(b\_2)
\end{array}
\right.
\end{array}$$

The combination of all previous steps gives us the following commutative diagram.

Let us recap a little what’s in diagram a little, to show that it is a justified construction:

* is a product, and so we may obtain , , , and ;
* is well-defined for the equivalence relation , and so we may obtain and ;
* is well-defined for the equivalence relation , and so we may obtain and ;
* is well-defined for the equivalence relation , and so we may obtain and ;

Therefore, we have the maps and . and are well-defined because they are compositions of well-defined maps. For and , we know that the projector and unique function part (of the universal property of quotients) are necessarily well-defined, otherwise they wouldn’t compose to a well-defined map. As for and , they are well-defined because they are projections, and projections are always well-defined.

Since and are well-defined, we can "quotient through" (i.e., use the universal property of quotient for) .

This gives us the following commutative diagram (compatible with the one above, but it’s too messy to represent both at the same time):

These are precisely our functions and , and they are unique, as per the universal property of quotients.

##### 5.11.b)

What we wish to prove now is that is the product of and . Said otherwise, we wish to show that, for all , and in the appropriate configuration, the following diagram commutes:

Using the fact that is already a product, and studying maps and from into and respectively, we can make the following commutative diagram:

The internal parts of the diagram (except and which are arbitrary, all other arrows are unique morphisms) force and to commute with the rest of the diagram: any pairs of maps and that make the diagram commute must necessarily be constrained by the existing morphisms (as the composition of unique morphisms). Any choice of maps and must respect and , for corresponding arbitrary (but inferred) and ; said otherwise, maps and in this configuration can only exist if they make the diagram commute. Therefore, we have shown that satisfies the universal property of the product of and , and we have (though it’s not shown on the above diagram, in order to improve legibility).

##### 5.11.c)

We know that elements that verify a common universal property are terminal objects (of the same kind) in some comma category. Here, both and are final objects in the bislice category $\text{\textbf{Set}}\_{A,B}$. Since they are both final objects in the same category, they are isomorphic.

Conclusion:

#### 5.12)

Define notions of fibered products and coproducts, as terminal objects of the categories , considered in Example 3.10 (cf. also Exercise 3.11), by stating carefully the corresponding universal properties. As it happens, **Set** has both fibered products and coproducts. Define these objects ’concretely’, in terms of naive set theory. [II.3.9, III.6.10, III.6.11]

##### 5.12.a)

We first define the fibered product (or "pullback"). We suggest the reader goes to check section 3.11.3 of this exercises solutions document for a refresher on the concept "fibered bislice category" . We will use the same notation, i.e., fixing two morphisms and (with common codomain) in a category .

This means, for any generic object of , we have the following diagram:

We first propose a candidate for what it means to be a "fibered product", which we will be able to show is a final object in . A fibered product of and over in a category is an object together with morphisms and such that for any object in and morphisms and such that , there exists a unique morphism such that the following diagram commutes:

We will now show that this object induces a terminal object in ; this boils down to showing that is final in . Let be an arbitrary object of . A morphism from to is a "raising" of a morphism such that and . This raising is unique if and only if is unique. Since is presupposed to verify the universal property, such an , if it exists, is indeed unique. Therefore, is final (*a fortiori* terminal) in .

##### 5.12.b)

We now define the fibered coproduct (or "pushout"). We suggest the reader goes to check section 3.11.4 of this exercises solutions document for a refresher on the "fibered bicoslice category" . We will use the same notation, i.e., fixing two morphisms and (with common domain) in a category .

This means, for any generic object of , we have the following diagram:

We first propose a candidate for what it means to be a "fibered coproduct", which we will be able to show is an initial object in . A fibered coproduct of and over in a category is an object together with morphisms and such that for any object in and morphisms and such that , there exists a unique morphism such that the following diagram commutes:

We will now show that this object induces an initial object in ; this boils down to showing that is initial in . Let be an arbitrary object of . A morphism from to is a "raising" of a morphism such that and . This raising is unique if and only if is unique. Since is presupposed to verify the universal property, such an , if it exists, is indeed unique. Therefore, is initial (*a fortiori* terminal) in .

##### 5.12.c)

(I wouldn’t have come up with this without Wikipedia as a hint...)

In **Set**, the fibered product (pullback) of and over , is a special subset of the cartesian product that registers some extra information, pertaining to the functions and . We write this object as

.

Concretely, this can also be expressed as

. This is the set of all pairs (of inputs) such that . Let us show this is the case with a simple example.

Let , , and . We define as , and as . We have and , therefore . The fibered product is then:

A simple way to verify this is to verify that for each pair in , , and that no such other pairs are missing.

##### 5.12.d)

https://math.stackexchange.com/questions/3021738/pushout-in-the-category-of-sets-proof

https://math.stackexchange.com/questions/2240882/understanding-an-example-of-a-pushout-in-mathbfset

In **Set**, the fibered coproduct (pushout) of and over is a special quotient of the disjoint union that registers some extra information, pertaining to the functions and . We write this object as

, where is an equivalence relation defined as , the equivalence closure of the relation

(i.e., two elements are equivalent if they are both the output of some common , or if there is any chain of such equivalences between them).

(We remind that the equivalence closure, if elements of the set are seen as vertices and the relation pairs as edges of a graph, can be thought of, visually, as "completing the cliques" of each connected component in the graph, including self-loops; hence the idea that if there is any path between two elements, then they are equivalent and should be directly linked. We also remind that this corresponds uniquely to a partition of the set; the elements of which, called cosets, are what allow us to do an algebraic quotient.)

The fibered coproduct is the set obtained by taking the disjoint union and identifying with if there exists such that and (and all identifications that follow to keep the equality relation an equivalence relation). There is no more concrete of a definition than this; it really boils down to identifying elements with a common preimage element through and in , via an equivalence relation.

Let us show this is the case with a simple example. We will keep the elements of and distinct in order to remove the visual clutter that comes with the and of the general disjoint union.

Let , , and . We define as , and as . We have:

* and , so ;
* and , so , and by closure, ;
* and , so ;
* and , so you might think that , however, since there is no such that and , we have ;

This information corresponds to the following partition of : . The fibered coproduct is then:

A way to verify this is to verify that each equivalence class is disjoint, and that all pairs of elements within an equivalence class are related by (by applying or where appropriate and drawing the graph of the relation).

## Chapter II)

### Section 1)

#### 1.1)

Write a careful proof that every group is the group of isomorphisms of a groupoid. In particular, every group is the group of automorphisms of some object in some category.

Let us first remind the definition of a groupoid: a groupoid is a category in which every morphism is an isomorphism.

(Side-note: by "group of isomorphisms", what Aluffi rather meant is "group of isomorphisms (of the single object) in a groupoid (that has a single object)". Otherwise, with multiple objects, things fail: for example, there are multiple identities, one per object, not universally applicable to each object. This is precisely why the notion of "groupoid" was invented, to extend the notion of group in such a way.)

Let be a group, i.e., some form of algebraic structure with a set of elements and a binary operation which is associative, unitary, and invertible. We want to show that there exists a groupoid such that is the group of isomorphisms of .

Let us define as follows:

* There is a single object in , and its elements are the elements of (we could call "", of course, but we’ll be distinct for pedagogy’s sake). Our goal is to prove that is isomorphic to , and thus is itself a group.
* For any element of , there is a unique morphism such that .
* Composition of morphisms is defined as follows: .
* There is an identity morphism , with the identity element of , and .
* It is immediate to see that these morphisms are associative since is associative. Take , , and in , for : .
* Every such morphism has an inverse, namely, which by definition of a group necessarily exists. It is easy to verify that .

This is a groupoid, because, it is a category (composition, associativity, identity) where every morphism is an isomorphism (every morphism has an inverse), and the group of isomorphisms of (the single-object category) , here , is precisely isomorphic to .

#### 1.2)

Consider the ’sets of numbers’ listed in §1.1, and decide which are made into groups by conventional operations such as and . Even if the answer is negative (for example: is not a group), see if variations on the definition of these sets lead to groups (for example, is a group, cf. §1.4). [§1.2]

I suppose Aluffi is referring to §I.1.1, and not §(II.)1.1. In there he mentions:

* : the set of natural numbers (that is, nonnegative integers);
* : the set of integers;
* : the set of rational numbers;
* : the set of real numbers;
* : the set of complex numbers.

Let us go through these sets one by one:

* : is not a group (needs negative numbers, has no solution), and neither is (e.g., has no solution in ).
* : is a group, but is not because it is not invertible (e.g., has no solution in ).
* : is a group, but is not because it is not fully invertible (e.g., has no solution in ). However, remove from and it becomes a group.
* : is a group, but is not because it is not fully invertible (e.g., has no solution in ). However, remove from and it becomes a group.
* : is a group, but is not because it is not fully invertible (e.g., has no solution in ). However, remove from and it becomes a group.

We can see that is not a group, but , , , and are all groups. Also, , , , , and are not groups. However, , , and are groups.

#### 1.3)

Prove that for all elements of a group .

Therefore, is a 2-sided inverse for , and since the inverse of an element in a group is unique,

#### 1.4)

Suppose that for all elements of a group ; prove that is commutative.

If , then every element is its own inverse, i.e., multiply by on the left (or right), and we have . Let be . Then, we have , which is the definition of commutativity. Therefore, is commutative.

#### 1.5)

The ’multiplication table’ of a group is an array compiling the results of all multiplications (with the value on each row being the left operand, and the value on each column being the right operand; of course the table depends on the order in which the elements are listed in the top row and leftmost column). Prove that every row and every column of the multiplication table of a group contains all elements of the group exactly once (like Sudoku diagrams!).

Another way to phrase this question is "prove that every element of the group can be reached in a single operation from any other (on the left, or on the right, both work)". Written in function notation, this means:

For every element , we can reach the identity with (on either side). Every element can then be reached from the identity by multiplying it (on either side). We just have to pick the same side both times to find either and . Since the group is closed under "multiplication", both these elements are guaranteed to exist.

This implies that every row and every column of the multiplication table of a group contains all elements of the group exactly once, since any element can be reached in a single multiplication (i.e., there as many possible inputs (1 sided operand) as possible outputs (i.e., applying an operand is an injection) and all outputs are reached (it’s also a surjection)). This also implies that the multiplication table is a "Latin square", which is a square array of symbols, each occurring exactly once in each row and exactly once in each column.

#### 1.6)

Prove that there is only one possible multiplication table for if has exactly 1, 2, or 3 elements. Analyze the possible multiplication tables for groups with exactly 4 elements, and show that there are two distinct tables, up to reordering the elements of . Use these tables to prove that all groups with elements are commutative.

These multiplication tables are usually called "Cayley tables".

##### 1.6.a) 1-element group

For the unique table of the trivial group (only 1 element); is necessarily its own inverse. Some examples of this group are or . This group is trivially commutative.

|  |  |
| --- | --- |
|  |  |
|  |  |

##### 1.6.b) 2-element group

For the unique table of the group with 2 elements; both elements are necessarily their own inverse (or else there would be no which is necessarily a self-inverse). Some examples of this group are , or (two-hour clock with addition). This group is commutative, since, like in the exercise 1.4, every element is its own inverse.

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |

##### 1.6.c) 3-element group

For the unique table of the group with 3 elements; the 3 elements cannot all be their own inverses, because if 2 elements are their own inverse, the third one cannot have an inverse. Another way of seeing this is the below: no matter the value given for each , this cannot be a Latin square (if we put two ’s then we have multiple inverses, so we can deduce that and are inverses, and then we ; if we put two ’s, we lose associativity (e.g.: ); etc).

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Instead, the 2 non-identity elements are necessarily inverses of each other. Some examples of this group are (where is a complex number called the third root of unity), or (three-hour clock with addition). This group is commutative because the identity necessarily commutes with everything, and inverses necessarily commute together, so here, all elements commute.

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

##### 1.6.d) 4-element groups

For the case where , we have two possibilities.

The first one is a table where every element is a self-inverse. This gives you a group that is isomorphic to (the Klein four-group, a torus made up of 2 two-hour clocks). Note that the permutations of the elements of the group give the same table, so there is only one table for this group up to reordering. It is commutative for the same reason as exercise 1.4.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

The second possibility where one element other than the identity is its own inverse, and the other two are mutual inverses. This gives you a group that is isomorphic to (a four-hour clock with addition, where and are mutual inverses, and and are their own inverse). Note that the permutations of the elements of the group give the same table, so there is only one table for this group up to reordering.

Since the identity and respective inverses commute, all that’s left is to check the commutativity of the self-inverse element with both of the mutual-inverses elements and . In this case, it is because there is "only one option left" in the Latin square that we get commutativity: e.g., cannot equal since they are not inverses, cannot equal since , cannot equal since , so we must have . The same reasoning can be applied to to get (hence ). With a relabelling, we apply this reasoning to to get and to to get (hence ). With this, we have proven commutativity.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

##### 1.6.e) Conclusion and remark

With that, we’ve proven that all groups of order are commutative.

(Note that the commutativity can also be "seen" in all these tables by the fact that they are their own transpose.)

#### 1.7)

Prove Corollary 1.11: Let be an element of finite order, and let . Then ( is a multiple of ).

By Lemma 1.10, if , then divides . Therefore, .

For the converse, we suppose . By definition of the order of a group element, , so . Hence, .

#### 1.8)

Let be a finite group, with exactly one element of order . Prove that .

I had some trouble, so I checked online and found this, meaning that the group probably needs to be not just finite but also abelian:

https://math.stackexchange.com/questions/2550052/let-g-be-a-finite-abelian-group-with-exactly-one-element-of-order-2-denoted

(A couple of notes: since , it also means that is the only self-inverse (involution) in . Also, the order of operation is not given: this gives us a hint that the property does not rely on the order of operations.)

Since is the only element which is a self-inverse of order , is the only self-inverse of order , we have that for all other elements a in , is not its own inverse. This means that the product of all elements of is a product of pairs of inverses, which cancel out to the identity (given commutativity), except for the self-inverse which remains. Therefore, .

#### 1.9)

Let be a finite group, of order , and let be the number of elements of order exactly . Prove that is odd. Deduce that if is even then necessarily contains elements of order .

(This would be trivial if we could use Lagrange’s theorem.)

Let us consider the set of elements of of order , , with . We know that for all , . This means that is the set of self-inverse elements (not counting ). This means that contains only pairs of inverses (with such pairs), and the self-inverse , so has its cardinal can be written , i.e., it is an odd number.

If is even, then it can be written as for some with . Consequently , which is odd, and since . Therefore, necessarily contains at least element of order .

#### 1.10)

Suppose the order of is odd. What can you say about the order of ?

We write since it is odd.

By Corollary 1.11, . Since is odd, both and must be odd.

#### 1.11)

Prove that for all in a group , . (Hint: prove that for all .)

Let .

By Lemma 1.10, is a divisor of .

By definition, . However, similarly to above, . So we have ; then, multiplying on the left by and right by , we get . By Lemma 1.10, is a divisor of .

Since the divisibility relation is an order relation (antisymmetric), the only time two numbers can be mutually divisors of each other is if we’re in the reflexive case (i.e., they are equal). Therefore, .

Applying this identity to the elements with , we have

#### 1.12)

In the group of invertible matrices, consider , . Verify that , , and . [§1.6]

We remind call that the neutral element for 2D square matrix multiplication is the identity matrix .

We will now prove by induction that .

Initiatialization:

True for .

Inheritance:

We suppose true for a specific , we’ll show that holds.

Therefore is true. This prives inheritance.

By induction, is true for all .

Therefore, there exists no such that , hence .

#### 1.13)

Give an example showing that is not necessarily equal to , even if and commute. [§1.6, 1.14]

We take the 6-hour clock , and , the equivalence class of 3 modulo 6. We have , so and . However,

#### 1.14)

As a counterpoint to Exercise 1.13, prove that if and commute, and (they are coprime), then . (Hint: let ; then . What can you say about this element?) [§1.6, 1.15, §IV.2.5]

First, we remark that for any , , so .

We have that , and we know that so we have .

Let . Then, given that , that and commute, and multiplying by on the right of both side, we have . Therefore, .

We now study and .

We have .

Similarly, (by our initial remark).

Therefore, divides both and . Since and are coprime, . Therefore , so is a multiple of and so is a multiple of . Since and are coprime, we can further say that is a multiple of their LCM/product (using, for example, the fundamental theorem of arithmetic).

Using Proposition 1.14 (i.e, if , then divides ), we can also tell that divides .

Putting both results together using the antisymmetry of the divisibility relation, we have .

#### 1.15)

Let be a commutative group, and let be an element of maximal finite order: that is, such that if has finite order then . Prove that in fact if has finite order in then divides . (Hint: argue by contradiction. If is finite but does not divide , then there is a prime integer such that , , with and relatively prime to , and . Use Exercise 1.14 (the fact that the order of the product of two elements with coprime order is equal to the product of their orders) to compute the order of .) [§2.1, 4.11, IV.6.15]

We suppose that is finite but does not divide . Using the fundamental theorem of arithmetic, if doesn’t divide , then it has at least one prime factor that does not have. Then, there is a prime integer such that , , with and (potentially composite), both independently coprime with , and .

We know from exercise 1.14 that if 2 elements and commute (which is always the case in a commutative group like here), if the two elements have coprime order, then .

The order of , is , assuming is finite (Proposition 1.13).

We study .

The order of is because divides .

Similarly, the order of is because divides .

Since is a commutative group, and commute. By Exercise 1.14, .

This is a contradiction with the fact that has *maximal* finite order. Therefore, our assumption that does not divide must be false.

Conclusion: if is an abelian group, has maximal finite order, and has finite order in , then divides .

### Section 2)

#### 2.1)

One can associate an matrix with a permutation , by letting the entry at be 1, and letting all other entries be 0. For example, the matrix corresponding to the permutation

(Aluffi seems to have made a mistake, writing instead of , so we fixed it here)

would be

Prove that, with this notation, for all , where the product on the right is the ordinary product of matrices.

We first notice that we can write down the formal expression of , in the dummy Kronecker delta style, is just the permutation of the columns of the identity : here . Therefore, using the Kronecker delta to match the indices, we have that the product of two matrices

as desired.

(): each term acts a Kronecker delta, in particular, the second term is non-null iff , which acts as our replacement in the following line.

#### 2.2)

Prove that if then contains elements of order .

We will use strong induction on .

Initialization: for , the case is trivial (the identity element has order 1). For , we saw that this group was necessarily isomorphic to , which has the identity, and one element of order 2.

Heredity: For some , we suppose that the property is true for all . We will show that it is true for . Keeping the last element fixed, we see that the portion of that can permute is isomorphic to : by our hypothesis, it thus has elements of order for all . We just need to show that we can find an element of order in .

We take the a cyclic permutation of all elements (which acts like a Caesar cypher of distance parameter ): each is mapped to to , and is mapped to . Let us prove that this permutation has order .

We can see that the cycle of this permutation is , which has length .

We now show that the order of any cyclic permutation is the length of the cycle. Let be a permutation group and a cycle in it, generated by a permutation . We denote elements of the cycle as with such that the cycle can be expressed as . We have that , trivially. Then, we have (meaning modulo plus ), and so on. We see that , and . Therefore, the order of is , the length of the cycle.

Therefore, we have shown that has cyclic permutations of length , and thus elements of order , which completes the proof.

#### 2.3)

For every positive integer , find an element of order in .

From the previous exercise, it is clear that any cyclic permutation of length is such an element.

#### 2.4)

Define a homomorphism by labeling vertices of a square, as we did for a triangle in §2.2. List the 8 permutations in the image of this homomorphism.

We first write the dihedral group as the group of symmetries of a square. We label the vertices of the square as , in a clockwise manner. We use matrices with these labels to represent the symmetries of the square, and we have that the elements of are:

Now, using the notation for permutations, we can write the corresponding permutations in as:

Note that there exist other permutations which are not in the image of this homomorphism, such as the transposition , which is not a symmetry of the square. This homomorphism is thus injective, but not bijective.

#### 2.5)

Describe generators and relations for all dihedral groups . (Hint: let be the reflection about a line through the center of a regular -gon and a vertex, and let be counterclockwise rotation by . The group will be generated by and , subject to three relations. To see that these relations really determine , use them to show that any product equals for some , with , .)

Two of the relations are very easy to see: (for any axial symmetry flip ) and (for any -rotation ). Another hint (given at the bottom of page 52, as done on an imaginary pentagon) tells us to study (hinting that it is equal to ).

We first note that since any axial symmetry is an involution. Geometrically speaking, we can see that the operation is a rotation by in the opposite direction to . We can thus write , and thus . This is our 3rd relation, which we can also rewrite as .

We can now write any element of as for , . Any power of reduces to (if the exponent is even) or (if the exponent is odd), any power of reduces to , and any term sandwiched between an and a terms can be changed to in order to reduce with the (or ) on the left, and the on the right. Through this algorithm, we can always get an element of the form for , .

#### 2.6)

For every positive integer construct a group containing two elements , such that , , and . (Hint: for , will do.) [§1.6]

We can take the dihedral group , and take the reflection and the rotation (as defined in the previous exercise). Taking and , we have that (by the first relation), (by the third relation) and (by the second relation).

As for the case with , we take the group and set ; their sum is the identity , which has order (technically, to stay in the picture, this is isomorphic to the group of symmtries of a "2-gon", a line, with either the identity, or a flip across its mediatrix).

#### 2.7)

Find all elements of that commute with every other element. (The parity of n plays a role.)

A pair of elements commute iff . Let be our test element. We want to have the property . We can write for , . We have that . Therefore, only elements such that commute, this is equivalent (by cancelling the eventual on both sides and aggregating the ’s) to , which is true iff or . Therefore, if is odd, only the identity commutes with all elements, and if is even, both the identity and the rotation of angle commute with all elements.

#### 2.8)

Find the orders of the groups of symmetries of the five ’platonic solids’.

https://en.wikipedia.org/wiki/Polyhedral\_group

The five platonic solids are the tetrahedron, the cube, the octahedron, the dodecahedron, and the icosahedron. We can find the orders of their symmetry groups by finding the number of symmetries of each solid.

TetrahedronL The tetrahedron has 4 vertices, 6 edges, and 4 faces (all equilateral triangles). We can rotate the tetrahedron by about an axis perpendicular through the center of a face. We can also flip the tetrahedron across a plane through an edge and the midpoint of its opposite edge.

(distinct combinations ? proof ?)

... TODO

#### 2.9)

Verify carefully that ’congruence mod ’ is an equivalence relation.

We define the relation as . We will show that this relation is reflexive, symmetric, and transitive.

Reflexivity:

Symmetry:

Transitivity:

Therefore, the relation is an equivalence relation.

#### 2.10)

Prove that contains precisely elements.

Each element of can be written uniquely as with and . Therefore, we have possible values for . Because congruence modulo equates two numbers in if their difference is a multiple of , we have that and are equivalent. Therefore, we can consider an equivalence class associated with to contain all elements of the form . Since we have precisely such equivalence classes, contains precisely elements.

#### 2.11)

Prove that the square of every odd integer is congruent to 1 modulo 8. [§VII.5.1]

We write an odd integer as for . We have that . We can see that is always even, because: if is even, then is even, and is even, so their sum is even; and if is odd, then is odd, and is odd, so their sum is even. Therefore, we can write for some , and thus , which is congruent to 1 modulo 8.

#### 2.12)

Prove that there are no integers such that . (Hint: by studying the equation in , show that would all have to be even. Letting , , , you would have . What’s wrong with that?)

We can write as with . Any solution to the first equation must thus respect the properties of the second equation. We study these properties by looking at the equation modulo 4.

Since 0 is even, this condition gives us that either two of the numbers have to be odd, or none of them are. However, looking back at the first equation, if is even, then , and , which means that and must be both even or both odd. If and are both odd (i.e., either congruent to or ), then and and their sum, supposedly equal to , is congruent to , which is a contradiction. In this case, both and must be even.

Now if is odd, then . Since is odd, only one of or must be odd. Without loss of generality, we choose that to be , in which case . Our "the three sum to 0" equation above then tells us that , so , which is a contradiction, because there is no number such that (the image of the squaring function is ). Therefore, cannot be odd.

We’ve thus shown that the only case not leading to a contradition is the case where all three numbers are even, and this applies to the original equation.

Letting , , , you would have , which is just a rewriting of our initial equation. After iteration this process until a maximal instance of ’s has been removed from the prime factorization of each number, we will eventually reach a point where either , , or (non-exclusive) is odd, which is a contradiction. Therefore, there are no integers such that .

This exercise is interesting because it teaches us that number theoretic equations that rely on odd and evenness can effectively be studied using .

#### 2.13)

Prove that if , then there exist integers and such that . (Use Corollary 2.5.) Conversely, prove that if for some integers and , then .

This is known as Bézout’s identity. We remind Corollary 2.5.: The class generates if and only if . So this exercise can also be reduced to proving that if and only if generates .

We suppose that , by Corollary 2.5, we have that generates . This means that for every , there exists such that . If we take , this means that for some . We can rewrite this as for .

We suppose that . We can write this as , which means that divides . Suppose and have a common prime factor : this prime factor would divide and , and thus would divide (equivalently its opposite ) and . However, this is a contradiction, because the only number that can divide two numbers that are off by is . Therefore, and have no common prime factors, and .

#### 2.14)

State and prove an analog of Lemma 2.2, showing that the multiplication on is a well-defined operation.

We remind Lemma 2.2: If and, then .

Our analogue is thus: and, then .

We now show that it is well-defined (there exists a commutative square diagram with an epimorphism from to in one direction, and with the respective multiplications in the other). We take and. This means that and for some . We have that . The first and last terms of this equation give us that , as needed.

#### 2.15)

Let be an odd integer.

* Prove that if , then . (Use Exercise 2.13.)
* Prove that if , then . (Ditto.)
* Conclude that the function is a bijection between and .

The number of elements of is Euler’s -function. The reader must proved that if is odd, then .

#### 2.16)

Find the last digit of . (Work in .)

We note that , , and , so the order of in is . Since multiplication is well-defined in , we have:

Thus, the last digit of this large number is .

#### 2.17)

Show that if , then if and only if .

We use the result from exercise 2.13: ; similarly, . We have that , so for some . We can rewrite this as . We can now substitute in Bézout’s identity: . We can now let and , and we have that .

#### 2.18)

For , define an injective function preserving the operation: that is, such that the sum of equivalence classes in corresponds to the product of the corresponding permutations.

We can define the function as , where is the permutation that maps to modulo (our Caesar cypher mentioned above, over the first elements, or any choice of elements really).

This function is injective because the permutation is uniquely determined by . This can immediately be seen by studying with which element the first element (which we can label 0) is replaced: it is replaced with (the -th element), leading to distinct outputs for possible inputs. More formally, if , then and are distinct permutations, as they map to and respectively, and implies that for any .

We can see that the sum of equivalence classes in corresponds to the product of the corresponding permutations through the following commutative diagram:

$$\begin{tikzcd}
\begin{array}{c} (\mathbb{Z}/d\mathbb{Z})^2 \\ ([a]\_d, [b]\_d) \end{array} & \begin{array}{c} \mathbb{Z}/d\mathbb{Z} \\ {[a+b]\_d} \end{array} \\
\begin{array}{c} S\_n^2 \\ (\sigma\_a,\sigma\_b) \end{array} & \begin{array}{c} S\_n \\ \sigma\_{a+b} \end{array}
\arrow["{+\_{\mathbb{Z}/d\mathbb{Z}}}", from=1-1, to=1-2]
\arrow["{(f,f)}"', from=1-1, to=2-1]
\arrow["f", from=1-2, to=2-2]
\arrow["{\circ\_{S\_n}}"', from=2-1, to=2-2]
\end{tikzcd}$$

#### 2.19)

Both and consist of elements. Write their multiplication tables, and prove that no re-ordering of the elements will make them match.

For , we have the elements which are coprime with .

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

For , we have the elements which are coprime with .

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

The two groups are not isomorphic, because the first group has elements of order and the other doesn’t. In fact, the former is isomorphic to , and the latter is isomorphic to (the Klein 4-group).

# Extra exercises by/for the group

### Chapter I) 1) Set notation)

Write the following in set notation (as a list of numbers, and algebraically):

* the set of all odd integers
* the set of all integers that are not multiples of 3
* the set of integers from 10 (included) to 20 (included)
* the set of integers from 10 (included) to 20 (excluded)
* the set of pairs of integers with both elements of the same value
* the set of triplets of real numbers that together sum to 1
* the set of pairs of positive real numbers that together sum to 1
* the set of -tuplets (for any ) of real number that together sum to 1
* the set of all natural numbers such that there exists at least one triplet of positive even numbers which are all different and which sum to that number.

Now take the sets in their algebraic notation, and represent them both as a list of numbers (as a logical sequence or just a couple of examples), and as a "description" of what they are:

### Chapter I) 3) Slices and coslices)

Provide a concrete example of a slice category and of a coslice category based on the category of real vector spaces , or its subcategory of finite real vector spaces. How does this relate to the transpose of a matrix, or of a product of matrices ?

# Notes

## Chapter 1, Section 1

Go check out the extra exercises on set notation.

## Chapter 1, Section 2

### On injections and surjections

#### Injections

Injections (which aren’t also surjections) have multiple left-inverses (post-inverses). Eg:

It is precisely the free element with no antecedent in (here, ) which leaves room for multiple choices, but doesn’t affect the overall inversion process.

#### Surjections

Surjections (which aren’t also injections) have multiple right-inverses (pre-inverses), called sections.

It is precisely the fact that there are multiple elements that map to the same element (here, and to ) which leaves room for multiple choices, but doesn’t affect the overall inversion process.

#### Cancellations

Function Cancellation Lemma: If a composition of functions cancels out, then the first of the pair is an injection, and the second of the pair is a surjection. Algebraically:

Corollary 1: any post-inverse of an injection is a surjection.

Corollary 2: any pre-inverse of a surjection is an injection.

Proof: Let be

a) Suppose is not an injection. This means:

However, with such an , we have:

This means that is an injection. Contradiction.

Conclusion: in this context, must be an injection.

b) Suppose is not a surjection. This means:

Since , that means that , this means that . Contradiction.

Conclusion: in this context, must be a surjection.

### On sections and fibers

Section example with a tangent bundle.

Consider the cylinder , and the function , the projection onto the circle. The cylinder is itself the space in which one can easily represent maps of . Each such map corresponds to a section.

Let be

We have

(TODO add diagrams for S1xR, g1 and g2)

A fiber is the preimage of a singleton. In the case of above, for every , is the copy of the real line on the cylinder that passes by .

(TODO add diagram)

### Alternative characterization of a bijection

" is bijective" ("every element of has a non-empty fiber" (surjection) and "every fiber is a singleton" (injection))

### On monomorphisms and epimorphisms

#### Failing the mono/epi condition

##### Example of failing the monomorphism definition for a non-injection

Monomorphism definition:

$$\text{$f : A \to B$ is a monomorphism}
\\ \Leftrightarrow \\
\forall Z \in \text{Obj}(\mathcal{C}), \;
\forall g\_1, g\_2 \in \text{Hom}(Z, A), \;
(f \circ g\_1 = f \circ g\_2 \Rightarrow g\_1 = g\_2)$$

The multiple choice of elements (here, and ) in which map to in is precisely what allows the overall composition to be equal, even when . This provides insight into a proof of " is a monomorphism implies that is an injection". If you suppose that is a monomorphism and not an injection, you can easily reach a contradiction, since you can use elements like and that both map to the same to construct a counter-example to the implication that defines a monomorphism.

##### Example of failing the epimorphism definition for a non-surjection

Epimorphism definition:

$$\text{$f : A \to B$ is an epimorphism}
\\ \Leftrightarrow \\
\forall Z \in \text{Obj}(\mathcal{C}), \;
\forall g\_1, g\_2 \in \text{Hom}(B, Z), \;
(g\_1 \circ f = g\_2 \circ f \Rightarrow g\_1 = g\_2)$$

The element in not being reached by is precisely that which provides the opportunity to build such that they compose to the same result with , since the output of for them doesn’t affect the overall composition. This provides insight into a proof of " is an epimorphism implies that is a surjection". If you suppose that is an epimorphism and not a surjection, you can easily reach a contradiction, since you can use elements like that are not reached by to construct a counter-example to the implication that defines an epimorphism.

#### Proofs of mono/inj and epi/surj equivalence

Let .

The parts which are "Injection => Monomorphism" and "Surjection => Epimorphism" both use the respective sided inverses to prove the implication.

The other parts use the following tautology to prove the implication by contradiction. "Suppose that and , show that it leads to a contradiction".

##### Injection => Monomorphism

Suppose that is an injection. It thus has post-inverses.

From there:

We conclude that is also a monomorphism.

##### Surjection => Epimorphism

Suppose that is a surjection. It thus has pre-inverses.

From there:

We conclude that is also an epimorphism.

##### Monomorphism => Injection

Suppose that is a monomorphism.

Suppose now that is not an injection. Algebraically, this means that:

We can construct and such that but , using such a pair . Thereby, we prove that is not an monomorphism and arrive at a contradiction.

(If is the empty set, being initial in , there is only 1 map into (the empty map) and always hold. Therefore, any counterexample to the epimorphism definition needs to have at least 1 element.)

Let .

Clearly, . However, we also have:

This means that is not a monomorphism: contradiction.

Conclusion: is an injection.

##### Epimorphism => Surjection

Suppose that is an epimorphism.

Suppose now that isn’t a surjection. Algebraically, this means that:

We can construct and such that but , using such an . Thereby, we prove that is not an epimorphism and arrive at a contradiction.

(If is the singleton set, being terminal in , there is only 1 map into and always hold. Therefore, any counterexample to the epimorphism definition needs to have at least 2 elements. We will however use a 3-element set, since it makes things more intuitive and pedagogical.)

Let .

Clearly, . However, since is the domain of , of , and of , we have:

This means that is not an epimorphism: contradiction.

Conclusion: is a surjection.

## Chapter 1, Section 3

### Example summary

* (3.2): Set, category of sets as objects and set functions as morphisms.
* (3.3): preorder (or order, or equivalence relation) over a (single) set, transformed into a category; elements of the set as objects, and elements of the preorder (which is a relation, hence a subset of the cartesian product) as morphisms.
* (3.4): the powerset with the inclusion operator, transformed into a category; elements of the powerset (i.e., subsets of the set) as objects, and inclusion relations as morphisms (this is just an example of a preorder / order / equivalence category seen in 3.3).
* (3.5): slice categories , categories which isolate a specific object of a given category , and studies the morphisms into that object; an object of is any morphism from any arbitrary objet into (not the homset itself !) and a morphism in (from to ) is a "raising" into of a morphism in that preserves composition on morphisms in (i.e., ).
* (3.6): combining examples 3.3 and 3.5, first start with an order category on the set (there is a morphism iff ), then select a specific object (here, ) then study all morphisms of the category into (so the relation for any ); the morphisms are then simply given by the transitivity of , i.e., ( corresponds to , meaning our transforming into , here, corresponds to is transformed into ).
* (3.7): coslice categories (morphisms out of a chosen object).
* (3.8): the category **Set** of pointed sets, a coslice category over **Set** and any singleton set . Objects in **Set** are regular sets, but with a unique distinguished element; morphisms are any set functions that map the domain’s distinguised element to the codomain’s distinguished element.
* (3.9): "bislice" and "bicoslice" categories, basically a similar construct as slice and coslice, but taking two objects of the starting category, and studying pairs of morphisms (from a common domain, resp codomain) into (resp from) this pair.
* (3.10): "fibered bislice" and "fibered bicoslice" categories, once again a similair construct, but this time taking two *morphisms* into a common set C (resp. from a common set C).

### On terminal and initial objects in **Set**

If $\empty$ is initial and is terminal, it is because a function in (in categorical terms) must always have an output for every input. Ie, in category theory, all functions are maps ("applications").

Said algebraically:

$$\forall A, B \in \text{Obj}(\bold{Set}), \;
\forall a \in A, \;
\forall f \in \text{Hom}(A, B), \;
\exists f(a) \in B$$

In the case of $\empty$ as the input set, and there is only one function $f: \empty \to Z$ for any : is the empty mapping. But any $Z \to \empty$ (expept $\empty \to \empty$) contains no mapping (since we’d necessarily be ignoring at least one element of ).

Similarly, in the case of the (unique up-to-isomorphism) singleton set as the output, you’d have multiple functions (precisely ) into it, if you could ignore some elements of the input set. However, if all elements of the input set are required, that leaves you with only one function possible from : the function which maps all elements to .

### Restrictions and extensions of functions, and its consequences on a function’s nature

8 possibilities to study, based on the following binary dichotomies:

* injection or surjection
* enlarging or restricting
* domain or codomain

Note that enlarging the domain sometimes implies enlarging the codomain, and restricting the codomain sometimes implies restricting the domain.

Legend: Yes, No, Depends

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | enlarge dom | restrict dom | enlarge cod | restrict cod |
| injection | D | Y | Y | Y |
| surjection | Y | D | N | Y |

Theorems:

A) if , then , the restriction of the function to the corresponding smaller codomain, is also an injection (resp. surjection).

B) if , then (with the the canonical injection of into its superset ), is also an injection (resp. is never a surjection).

C) if , then , we have that is also an injection. However, one can construct such that stops being a surjection.

D) if , then , we have that is also a surjection. However, one can construct such that stops being a injection.

Proof: TODO

### On the morphisms of slice and coslice categories

Given a base category , and some set we wish to study the homsets of the slice (resp. coslice) category (resp. . **These homsets might be empty, or have more than one element.**

We remind that slice categories consider *morphisms to A* as their *objects* (written as for any ), while coslice categories consider *morphisms from A* as their *objects* (written as for any . *Morphisms*, of the form (resp. ) in a slice category (resp. coslice category ) map such objects to one another if and only if there exists a morphism in (the base category!) such that (resp. ).

#### Example of no for a slice category

For example, note that there exist pairs of morphisms and between which there is no morphism that exists in the slice category. One such example we can make is in . If we take the maps:

There exists no map such that the following diagram commutes (since the output of will always be null in its second coordinate, and the output of will always be null in the first):

#### Example of no for a coslice category

TODO add potato diagram

Take , , and . Since , there is no possible case in which (even if is epi).

#### Example of multiple

TODO add potato diagram

Another example, this time in a coslice category, with , we take $\text{\textbf{Set}}\_A$. We take , . For and there exists two elements in . These originate from two functions in **Set**: and . The free element in which is not in offers a degree of freedom.

#### Explanations on the conditions for

In the first example, what causes the issue is the fact that the images of and in are distinct.

In the second example, what causes the issue is the fact that we’ve reduced our common domain to an insufficient intermediary object .

The "element which isn’t mapped to offers a degree of freedom and thus breaks unicity" in the last example should remind you of the notes on surjections. Indeed, when comparing and , we’re in a context which is reminiscent of the definition of an epimorphism.

We can be assured of being able to cancel iff is an epimorphism, and thus , if it exists, is unique. The proof for slice categories is similar, but given the reverse orders, uses monomorphisms.

More generally:

* a) in a slice category, there will be no if (proof ? and in more general categories than concrete categories ? TODO: iff ?)
* b) in a coslice category, there will be no if (proof ? and in more general categories than concrete categories ? TODO: iff ?)
* c) in a slice category, there is at most a single iff is mono
* d) in a coslice category, there is at most a single iff is epi

(see perhaps https://ncatlab.org/nlab/show/over+category and https://ncatlab.org/nlab/show/under+category )

## Chapter 1, Section 4

### Notes on counterintuitive rules

* in some categories (such as with ; or **Ring**), "mono and epi" does not imply "iso"
* in every *abelian* category, we have that "iso epi and mono" (and though **Set** is not abelian, the property still holds)
* while in **Set**, a function is an epimorphism (surjective) iff it has a pre-inverse, in **Grp**, some epimorphisms do not have right inverses.

## Chapter 1, Section 5

### Initial and terminal objects

* there are categories without either initial or terminal objects, such as the preorder category of with .
* there are categories with multiple initial or terminal objects (for example, in **Set**, every singleton set is a terminal object); however, these are respectively unique up to isomorphism
* any object which is both initial and terminal is called a zero object.

### Universal properties

#### "Normal" universal properties

Verbatim: "The most natural context in which to introduce universal properties requires a good familiarity with the language of functors, which we will only introduce at a later stage. [...] We say that a construction satisﬁes a universal property (or: ’is the solution to a universal problem’) when it may be viewed as a terminal object of a category."

Then: "The declaration/explanation of a universal property generally follows the pattern ’object X is universal with respect to the following property: for any Y such that..., there exists a unique morphism Y → X such that...’; this explanation hides the deﬁnition of an accessory category, and the statement that X is terminal."

This is a complicated way to say: there is some construct to decompose a morphism which is "universal" (always exists) and reduces the rest of the information of the morphism into something "unique" (hence terminal object of some subcategory).

#### Dual universal properties