

Functions and Relations

Functions

A **function** is a rule or mapping from one or more arguments to a "result" or "value" or "output". If the function is defined only on members of a set Δ , we can say it's (at least) a **partial function** on the **domain** Δ . If it's defined on *all* the members of Δ , then it's a **total function** on the domain Δ . When we say just "function" without qualification, we're normally talking about total functions.

There's another use of the term "domain" in the literature we'll be discussing, when we talk about the "domain" of a quantifier. These uses are related, but they aren't obviously and straightforwardly derivable from each other.

The **codomain** of a function f is a set Γ such that all of the values of f for any arguments are members of Γ . (I'll also say that f maps its arguments "into" Γ .) This doesn't exclude the possibility that there are some members of Γ that aren't the value of f for any argument. The particular subset of Γ whose members all *are* the value of f for some argument I'll call the **range** of f .

When f is function with domain Δ and codomain Γ , we'll write that $f: \Delta \rightarrow \Gamma$. When $d \in \Delta$ is such that $f(d) = c$, for some $c \in \Gamma$, we call c the **image** of d under f . But most often, I will tend to say instead that c is the "value" of f at d , or the "result" of f for argument d .

There is a lot of variation in how the term "range" is used. Some authors use it to mean what I'm calling the function's codomain. And some authors talk not only about the "image of d under f ", but also about "the image of f " itself, by which they mean what I'm calling f 's range.

If $\Theta \subseteq \Gamma$ is a subset of f 's codomain, then the **preimage** of Θ is the largest subset of f 's domain such that f maps every member of that set into Θ . For example, if $f: \mathbb{Z} \rightarrow \mathbb{N}$ maps each z to z^2 , then the preimage of $\{4, 9\}$ will be $\{-2, 2, -3, 3\}$.

I don't think we'll need to make use of this notion. I'm only mentioning it here for reference, in case you come across it and wonder what it means.

When f takes more than one argument, we'll say that its domain is a set of tuples, like $\Delta_1 \times \Delta_2$, and we'll write that $f: \Delta_1 \times \Delta_2 \rightarrow \Gamma$. This is not the only way to handle multiple arguments, but it is the style of handling them that you'll find in the literature we'll be reading.

Finally, the **graph** of a function f with domain Δ and codomain Γ is the set $\{(d, c) \mid d \in \Delta \text{ and } c \in \Gamma \text{ and } f(d) = c\}$. Sometimes I slip and call this the function's "extension", but in fact the standard terminology is to call this the function's "graph" and to reserve the term "extension" for *predicates*. (The graph of a unary function f *may* be the same set of pairs as the

extension of some dyadic predicate "F".)

Some authors identify functions with their graphs. This raises two sets of issues. One is similar to the kind of issue raised by the identifications of (a, b) with $\{\{a\}, \{a, b\}\}$ and of (a, b, c) with $(a, (b, c))$. There could be an intelligible argument about whether these objects should be identified, or rather whether they should instead be thought of as having some intimate (perhaps unique) correspondence.

A second kind of issue is different: this is that some authors, in certain contexts, want to distinguish between functions whose codomains differ, even if the functions have the same graph. (Similar things could be said about the domains, if the functions are partial.) That is, they may want to distinguish the function that maps every $z \in \mathbb{Z}$ to z^2 and whose codomain is the set of natural numbers \mathbb{N} , and the corresponding function whose codomain is the set of all integers \mathbb{Z} , even though none of the members of $\mathbb{Z} - \mathbb{N}$ are the value of this function for any argument.

I don't expect the merits of saying this to be obvious at this point in our studies. Like a number of other debates, I'm just flagging this so that you're aware that there are choices being made, about which arguments can be had. I think we can avoid taking a stand on the question whether there can be distinct functions with the same graph.

Functions cannot be "one-to-many." For a single argument (or tuple of arguments), the function must have only a single value. (Or if it's a partial function, the function must only have a single value if it's defined for that argument.) Of course, the single value could be a set. So if you found yourself in a situation where you wanted something like a one- Δ -to-many- Γ 's function, you might scratch your itch by instead working with a function whose values are sets of Γ 's.

Functions can however be "many-to-one." For instance, consider the height function. Perhaps Sam is taller than me, but Vera and I are the same height. Then the "height-in-centimeters" function would map Sam and me to distinct numbers, but would map me and Vera to the same number. That is, it sometimes maps distinct arguments (me and Vera) to the same value (whatever our common height in centimeters happens to be).

Some functions are not many-to-one. These functions always map distinct arguments to distinct values. For instance, let M be the set of males who have children, and P the set of people, and let $f: M \rightarrow P$ be the function that maps a member of M to his firstborn child. Assuming no genetic engineering or special fertilization techniques, no two males have the same firstborn child. Functions of this sort, that always map distinct arguments to distinct values, are known as **one-to-one**. Another term sometimes used for this is that the functions are "injective" or "an injection."

Whether a function is one-to-one may depend on what you take its domain to be. For example, if we considered the function "firstborn child" to be defined not on the set of male parents, but on the set of male or female parents, then it would no longer be one-to-one, because some distinct parents have the same firstborn child.

Notice that the range of the "firstborn child" relation is smaller than what I designated as its codomain. Some members of P are not anyone's firstborn child. When this is *not* the case---when every member of the codomain is the value of *some* argument to that function---then we say that the function is (not merely into but) **onto** that codomain. Another term sometimes used for this is that the function is "surjective."

Functions can be one-to-one without being onto, and they can be onto without being one-to-one. But they can also be both. Functions that do have both of these properties are called **one-to-one correspondences**. Another term used for this is that the function is "bijective" or "a bijection."

The terms "injective" and "surjective" can be applied to partial functions as well as to total functions. Generally, though, the term "bijection" is restricted to *total* functions.

As we'll discuss later, when a function is bijective, then its domain and codomain have to have the same size.

[Wikipedia](#) has some diagrams of functions having and lacking some of these features. So too does a Partee reading I assigned.

Here's another HOMEWORK EXERCISE:

13. Consider the functions described below, where $x \in X$ and $y \in \Psi$. Is the function injective? is it surjective?

- a. $f(x, y) = x$
- b. $f(x, y) = (y, x)$
- c. $f(x) = (x, y_0)$, for some fixed $y_0 \in \Psi$

A **permutation** is a bijective function from a set to itself. You can think of it as a way to shuffle the members of the set around. The members of the set don't need to be *ordered* in any way, to be permutable in this sense.

HOMEWORK EXERCISES:

14. Let A be a finite set, and f be a function from A into A . (a) Explain why f 's being injective implies it is also surjective. (b) Explain why f 's being surjective implies it is also injective.

15. Are (14a) and (14b) true when A is infinite? If not, give counterexamples.

One special kind of permutation maps every member of its domain to that very same member. This is known as the **identity function** on that domain.

Suppose we have two functions $f: \Delta \rightarrow \Gamma$ and $g: \Gamma \rightarrow P$. For example, f might be the function that maps me to my height in centimeters and g be the function that maps each number n to n^2 . Then there is a function with the domain Δ and codomain P which takes an argument $d \in \Delta$, first applies f to it, and then applies g to the result. In our example, this would be the function that maps me to (my height in centimeters)². This function is called **the composition of f and g** and is written as $g \circ f$. That is:

$$(g \circ f)(d) = g(f(d))$$

Some more HOMEWORK EXERCISES:

16. If *wife* is the function that maps a married individual to his (or her) wife, and *father* is the function that maps a married woman to her father, what is another name in English for the function $\text{father} \circ \text{wife}$?
17. When we're talking about the composition of more than two functions, sometimes we'll leave off the parentheses. Should $h \circ g \circ f$ be understood as $h \circ (g \circ f)$ or as $(h \circ g) \circ f$? Justify your answer.

If f 's codomain is the same set as its domain, then the values of f for some argument will also themselves be suitable arguments for f . So we can talk about $f \circ f$. And we can talk about $f \circ f \circ f$ and so on. Sometimes we use this shorthand notation:

...

$$f^3(d) = (f \circ f \circ f)(d) = f(f(f(d)))$$

$$f^2(d) = (f \circ f)(d) = f(f(d))$$

$$f^1(d) = f(d)$$

$$f^0(d) = I_{\Delta}(d) = d, \text{ where } I_{\Delta} \text{ is the identity function for } f\text{'s domain } \Delta$$

Note that this is very different from a convention you may have learned when doing trigonometry: $\sin^2 x = (\sin x)^2$, not $\sin(\sin x)$.

Suppose a function $f: \Delta \rightarrow \Gamma$ is a bijection. Then because the function is one-to-one, every value in Γ in f 's range is such that there is a unique member of Δ that f maps to it. And because the function is onto Γ , every value in Γ is in f 's range. So we can talk about the **inverse** of function f , written as $f^{-1}: \Gamma \rightarrow \Delta$, and defined such that:

$$\forall c \in \Gamma (\forall d \in \Delta (f^{-1}(c) = d \text{ iff } f(d) = c))$$

We've just said that all bijections have inverses; in fact, they have unique inverses. The converse is also true: any function with an inverse is a bijection.

HOMEWORK EXERCISES:

18. If $f: \Delta \rightarrow \Gamma$ is a bijection, (a) what is $f^{-1} \circ f$? (b) What is $f \circ f^{-1}$? It should be clear what the domain and codomain of your answers are.
19. If f and g are both bijective, must $g \circ f$ be bijective? Defend your answer.
20. (a) If $g \circ f$ is bijective, must f be injective? (b) must g be injective? (c) must f be surjective? (d) must g be surjective? Defend your answers.

If we have a function $f: \mathbb{Z} \rightarrow \mathbb{N}$, such that $f(z) = z^2$, some other ways to specify which particular function f is are:

$$f: z \mapsto z^2$$

$$f = \lambda z(z^2)$$

We'll aim to discuss this λ notation some more at the end of the term. If we have a function that instead takes two arguments, such as $g(y, z) = y + z$, then we might write:

$$g = \lambda(y, z)(y + z)$$

You may also encounter this kind of notation:

$$\lambda yz(y + z)$$

That is shorthand for:

$$\lambda y(\lambda z(y + z))$$

and it not quite the same as $\lambda(y, z)(y + z)$. It's a different style for defining an operation on multiple arguments. (It's called the "curried" style, after the logician [Haskell Curry](#).) The function $\lambda(y, z)(y + z)$ is defined only on pairs of numbers. It is not defined for single numbers. The function $\lambda y(\lambda z(y + z))$ on the other hand is defined for single numbers. The result of applying this function to the argument 2 is a function $\lambda z(2 + z)$ that maps arguments z to $2 + z$. These are somewhat different ideas, and in some contexts that difference is very important.

For the most part, however, we will just work with functions that take multiple arguments in the style of $\lambda(y, z)(y + z)$; and we will avoid the λ notation for this until we return to the topic of " λ -abstraction" later in the semester (if we get that far). We'll instead describe that function in the manner I began with: $g(y, z) = y + z$.

One last bit of vocabulary about functions: when a function takes a single argument, we call it a "unary" function. When it takes two arguments, "binary." When it takes three arguments, "ternary." And so on. The general property is called the function's "arity." Sometimes I may slip and talk of "monadic" and "dyadic" functions, and the function's "adicity." But in fact the dominant usage is that we talk about functions as having "arity"s and it's instead *predicates* that are monadic and dyadic and have adicities.

HOMEWORK EXERCISE:

21. Which of the following mathematical symbols express unary functions? Which express binary functions? Which don't express functions at all?
- +
 - <
 - \cup
 - \subseteq
 - sin

Relations

Whereas we talk of functions' *having a value* for a given argument (or for given arguments, which we're thinking of as a single tuple), a relation is instead something we talk about as *holding* (or not holding) between certain arguments. The most familiar kinds of relations are binary relations, like the relation that holds between x and z when x is a parent of z . But there are also ternary relations, like the relation

that holds between x and y and z when y is intermediate in size between x and z . And quaternary relations, and so on.

Notice that unlike functions, relations *are* permitted to be one-to-many. One x can stand in the parent relation to many z s. But some binary relations are such that whenever they hold between an x and z , they don't also hold between x and a distinct z' . These relations are called "functional." This generalizes to the ternary case like this: if the relation holds between x and y and z , it doesn't also hold between x and y and a z' distinct from z .

There is a close affinity between functions and functional relations. If you have a $(n + 1)$ -ary functional relation r , you can define the n -ary function which maps an n -tuple (x_1, x_2, \dots, x_n) to a value z iff r holds between x_1, x_2, \dots, x_n and z . (Though in general, you cannot rely on this being a *total* function; since r may not hold between every x in the domain you're considering and another object.) And vice versa: given the function, you can define the corresponding functional relation.

The notion of a relation is more general though, since we can have $(n + 1)$ -ary relations to which no n -ary function corresponds, like the parent relation. In general, there is no function that returns, for any x , the z that x is a parent of, since there may be several such z s. (Restricted to certain domains of x s, there *may* be such a function; but in general there won't be.)

Another way an $(n + 1)$ -ary relation could be associated with a function is with the $(n + 1)$ -ary (rather than n -ary) function that maps a tuple $(x_1, x_2, \dots, x_n, z)$ to the truth-value **True** if r holds between x_1, x_2, \dots, x_n and z , and to the truth-value **False** otherwise.

Vocabulary like "domain", "injective", "composition", and so on is applied to relations in a way analagous to the way it's applied to functions. x stands in the composition of relation r_1 and relation r_2 to z iff there's some y such that x stands in r_1 to y and y stands in r_2 to z .

The **inverse** of a relation r is defined to be the relation that holds between z and x iff r holds between x and z . Unlike functions, every relation has an inverse.

Sometimes this is instead called the "converse" of the relation. The "inverse" terminology better parallels the use of "inverse" when we talk about functions. The "converse" terminology better parallels our talk about conditionals: $C \supset A$ is the converse, not the inverse, of $A \supset C$.

HOMEWORK EXERCISE:

22. Unlike with functions, the composition of a relation and its inverse need not be an identity. Give an example that demonstrates this.

Later we will come back to the notion of relations and discuss special categories of relations in more detail, especially the relations that we can think of as "ordering" their arguments. For the moment, I'll just introduce a few pieces of vocabulary describing some binary relations, which I trust you'll already be familiar with:

A binary relation r is **reflexive** iff it holds between all of its arguments and themselves. This

is just a rough definition, we'll probably want to refine it in some way. For consider relations that hold between every male parent and himself, and may also hold between him and his children, but don't hold between any female parents and anything / anyone else. Should we count this relation as reflexive? We might say it's reflexive on the domain of male parents, but not reflexive on the domain of parents in general. Or we might count a relation as reflexive iff whenever it holds between x and anything, it also holds between x and x . But it's permitted not to hold between some members of its domain and anything else. (This would be a "partial relation", akin to the notion of a "partial function".)

When a relation fails to be reflexive, say that it's "not reflexive." There's also a technical notion of a relation's being **irreflexive**. This means that the relation *never* holds between x and x .

A binary relation r is **symmetric** iff whenever it holds between x and y , it also holds between y and x .

When a relation fails to be symmetric, say that it's "not symmetric." There are also technical notions of a relation's being **asymmetric** and its being **anti-symmetric**, which we'll explain later. They are not the same as the relation's failing to be symmetric. I don't think all of this vocabulary is very well-chosen. But unfortunately it is very well-entrenched.

A binary relation r is **transitive** iff whenever it holds between x and y and between y and z , then it also holds between x and z .

When a relation fails to be transitive, say that it's "not transitive." There's also a technical notion of a relation's being **intransitive**. This means that whenever there are x , y , and z such that the relation holds between x and y and also between y and z , it *never* holds between x and z (that is, for no x , y , and z does that obtain). Partridge gives the example of the is the mother of relation among human beings.

Observe that \subseteq is a transitive relation, and that \in is not a transitive relation, but neither is it intransitive.

HOMEWORK EXERCISES:

23. We talked about binary and ternary relations. What if there could be a *unary* relation? What would that be like? What would you call such a thing?
24. What if there could be a *nullary* relation, that is, a relation whose arity was 0? What would that be like? How many such relations could there be? What would you call it or them?

More on functions, relations, and sets

We described two ways in which you could "translate" between relations and functions. Earlier, we also described a translation between a function and the function's graph, which is a set. There is a similar translation between a relation and its graph. The graph of an n -ary function is a set of $(n + 1)$ -tuples, and that would also be the graph of an $(n + 1)$ -ary relation (not an n -ary relation).

So some sets, those whose members are tuples, can be "translated" into corresponding functions and/or

relations. If the set contains some tuples $(x_1, x_2, \dots, x_n, z)$ and $(x_1, x_2, \dots, x_n, z')$, where $z \neq z'$, then there is no corresponding n -ary function, but there is still an $(n + 1)$ -ary relation.

There's a different way to translate between sets and functions, akin to our second translation between relations and functions. And this method applies to any set. We say that a set Γ , whose members may be tuples or not, corresponds to a function $f: \Gamma \rightarrow \mathbf{2}$, where $\mathbf{2}$ is the set of truth-values, such that $f(g) = \mathbf{True}$ if $g \in \Gamma$, and $f(g) = \mathbf{False}$ otherwise. This is known as Γ 's **characteristic function**.

Given all these ways of moving back and forth between sets and functions and relations, it shouldn't be surprising that many authors will propose to reduce or define some of these notions in terms of others. As I've suggested a few times, even if those proposals are correct, we might still reasonably hesitate to build them into our *introductory definitions* of the notions. And thus I've refrained from doing that. Nothing we go on to discuss will turn on whether functions, relations, and sets are three different notions, or they should all be defined in terms of sets, or in terms of functions, or a generalization of one of these notions, or anything else.

All contents copyrighted by jim.pryor@nyu.edu
URL: <http://www.jimpryor.net/teaching/courses/logic/functions.htm>
Updated: 26 Jun 2014 1:11 PM