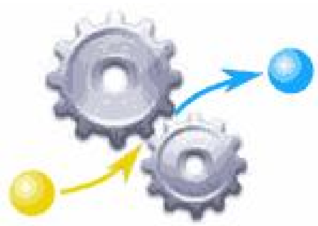
What is a Function?

A function relates an input to an output.



It is like a machine that has an input and an output.

And the output is related somehow to the input.

f(x)

"**f(x) = ...** " is the classic way of writing a function.
And there are other ways, as you will see!

Input, Relationship, Output

I will show you many ways to think about functions, but there will always be three main parts:

- The input
- The relationship
- The output

Example: "Multiply by 2" is a very simple function.

Here are the three parts:

Input	*Relationship*	Output
0	× 2	0
1	× 2	2
7	× 2	14
10	× 2	20
...	...	...

For an input of 50, what would be the output?

Some Examples of Functions

- x^2 (squaring) is a function
- x^3+1 is also a function

- Sine, Cosine and Tangent are functions used in trigonometry
- and there are lots more!

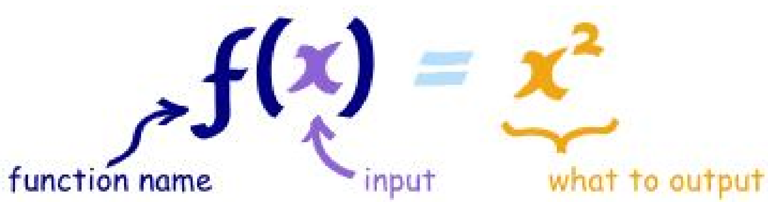
But we are not going to look at specific functions ...
... instead we will look at the **general idea** of a function.

Names

First, it is useful to give a function a **name**.

The most common name is "***f***", but you can have other names like "***g***" ... or even "***marmalade***" if you want.

But let's use "f":



You would say *"f of x equals x squared"*

what goes **into** the function is put inside parentheses () after the name of the function:

So ***f(x)*** shows you the function is called "***f***", and "***x***" goes **in**

And you will often see what a function does with the input:

f(x) = x² shows you that function "***f***" takes "***x***" and squares it.

Example: with **f(x) = x²**:

- an input of 4
- becomes an output of 16.

In fact we can write **f(4) = 16**.

The "x" is Just a Place-Holder!

Don't get too concerned about "x", it is just there to show you where the input goes and what happens to it.

It could be anything!

So this function:

$$f(x) = 1 - x + x^2$$

Would be the same function if I wrote:

- $f(q) = 1 - q + q^2$
- $h(A) = 1 - A + A^2$
- $w(\theta) = 1 - \theta + \theta^2$

It is just there so you know where to put the values:

$$f(\mathbf{2}) = 1 - \mathbf{2} + \mathbf{2}^2 = 3$$

Sometimes There is No Function Name

Sometimes a function has no name, and you might just see something like:

$$y = x^2$$

But there is still:

- an input (x)
- a relationship (squaring)
- and an output (y)

Relating

At the top I said that a function was **like** a machine. But a function doesn't really have belts or cogs or any moving parts - and it doesn't actually destroy what you put into it!

A function ***relates*** an input to an output.

Saying "**f(4) = 16**" is like saying 4 is somehow related to 16. Or 4 → 16

Example: this tree grows 20 cm every year, so the height of the tree is ***related*** to its age using the function ***h***:

***h*(age) = age × 20**

So, if the age is 10 years, the height is:

h(10) = 10 × 20 = 200 cm

Here are some example values:

age	**h(age) = age × 20**
0	0
1	20
3.2	64
15	300
...	...

What Types of Things Do Functions Process?

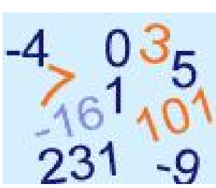
"Numbers" seems an obvious answer, but ...

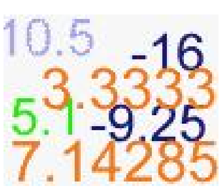
... **which** numbers?

For example, the tree-height function ***h*(age) = age×20** makes no sense for an age less than zero.

... it could also be letters ("A"→"B"), or ID codes ("A6309"→"Pass") or stranger things.

So we need something **more powerful**, and that is where sets come in:





A set is a collection of things.

Here are some examples:

Set of even numbers: {..., -4, -2, 0, 2, 4, ...}
Set of clothes: {"hat","shirt",...}
Set of prime numbers: {2, 3, 5, 7, 11, 13, 17, ...}
Positive multiples of 3 that are less than 10: {3, 6, 9}

Each individual **thing in the set** (such as "4" or "hat") is called a **member**, or **element**.

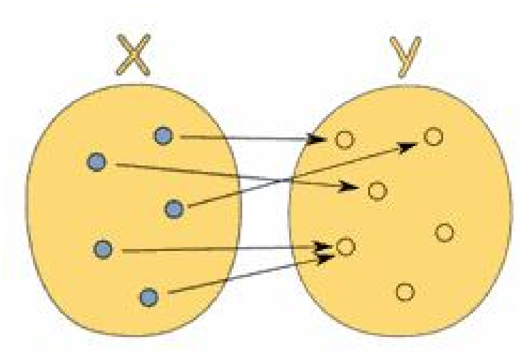
So, a function takes **elements of a set**, and gives back **elements of a set**.

A Function is Special

But a function has **special rules**:

- It must work for **every** possible input value
- And you can only have **one relationship** for each input value

This can be said in one definition:



Formal Definition of a Function

A function relates **each element** of a set with **exactly one** element of another set (possibly the same set).

The Two Important Things!

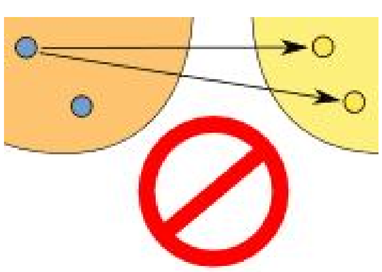
1. "...each element..." means that every element in **X** is related to some element in **Y**.

   We say that the function ***covers*** **X** (relates every element of it).

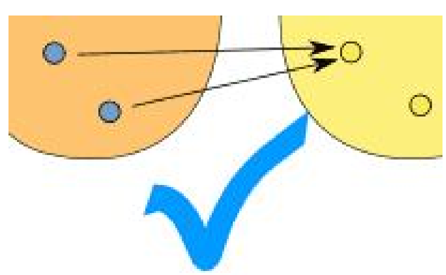
   (But some elements of **Y** might not be related to at all, which is fine.)

2. "...exactly one..." means that a function is ***single valued***. It will not give back 2 or more results for the same input.

So "f(2) = 7 **or** 9" is not right!



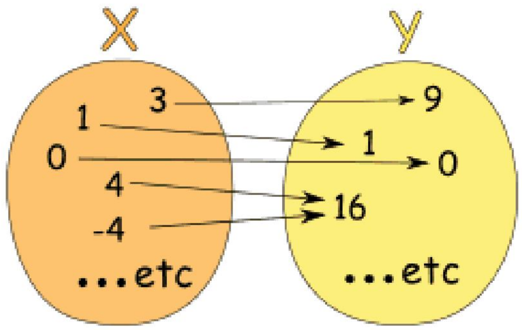
(one-to-many)
This is **NOT** OK in a function



(many-to-one)
But this **is** OK in a function

If a relationship does not follow those two rules then it is **not a function** ... it would still be a relationship, just not a function.

Example: The relationship $x \to x^2$



Could also be written as a table:

X: x	Y: x^2
3	9
1	1
0	0
4	16
-4	16
...	...

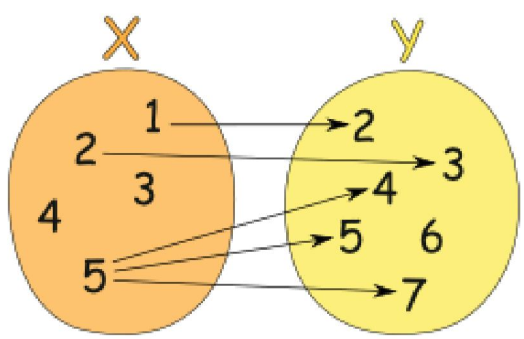
It is a function, because:

- Every element in X is related to Y
- No element in X has two or more relationships

So it follows the rules.

(Notice how both **4** and **-4** relate to **16**, which is allowed.)

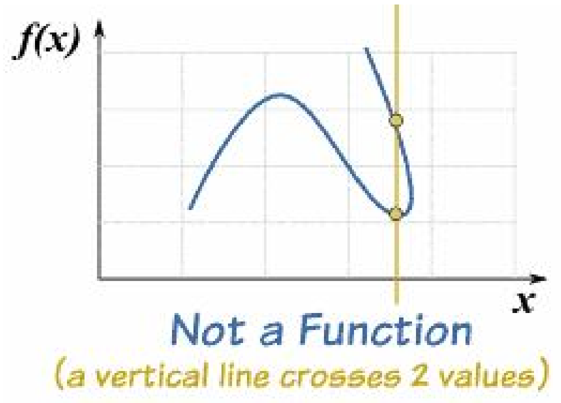
Example: This relationship is **not** a function:



It is a **relationship**, but it is **not a function**, for these reasons:

- Value "3" in X has no relation in Y
- Value "4" in X has no relation in Y
- Value "5" is related to more than one value in Y

(But the fact that "6" in Y is not related to does not matter)



Vertical Line Test

On a graph, the idea of **single valued** means that no vertical line would ever cross more than one value.

If it **crosses more than once** it is still a valid curve, but it would **not be a function.**

Some types of functions have stricter rules, to find out more you can read Injective, Surjective and Bijective

Infinitely Many

My examples have just a few values, but functions usually work on sets with infinitely many elements.

Example: $y = x^3$

- The input set "X" is all Real Numbers
- The output set "Y" is also all the Real Numbers

I can't show you ALL the values, so I just give a few as an example:

X: x	Y: x^3
-2	-8
-0.1	-0.001
0	0
1.1	1.331
3	27
and so on...	and so on...

Domain, Codomain and Range

In our examples above

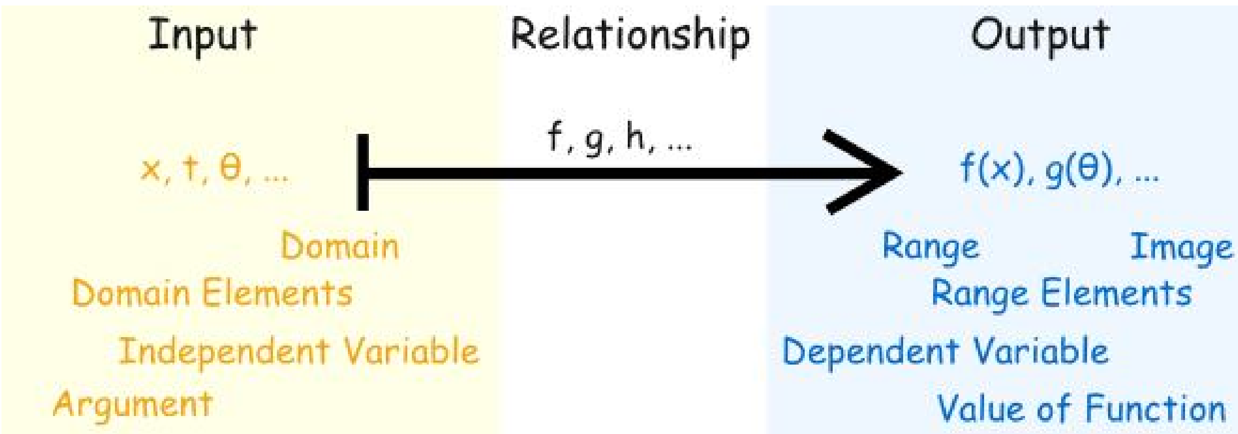
- the set "X" is called the **Domain**,
- the set "Y" is called the **Codomain**, and
- the set of elements that get pointed to in Y (the actual values produced by the function) is called the **Range**.

We have a special page on Domain, Range and Codomain if you want to know more.

So Many Names!

Functions have been used in mathematics for a very long time, and lots of different names and ways of writing functions have come about.

Here are some common terms you should get familiar with:



Example: with $z = 2u^3$:

- "u" could be called the "independent variable"
- "z" could be called the "dependent variable" (it **depends on** the value of u)

Example: with **f(4) = 16**:

- "4" could be called the "argument"
- "16" could be called the "value of the function"

Ordered Pairs

I said I would show you many ways to think about functions, and here is another way:

You can write the input and output of a function as an "ordered pair", such as (4,16).

They are called **ordered** pairs because the input always comes first, and the output second:

(input, output)

So it looks like this:

(**x**, **f(x)**)

Example:

(4,16) means that the function takes in "4" and gives out "16"

Set of Ordered Pairs

A function can then be defined as a **set** of ordered pairs:

Example: **{(2,4), (3,5), (7,3)}** is a function that says

"2 is related to 4", "3 is related to 5" and "7 is related 3".

Also, notice that:

- the domain is **{2,3,7}** (the input values)
- and the range is **{4,5,3}** (the output values)

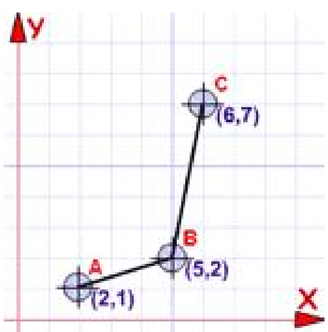
But the function has to be **single valued**, so we also say

"if it contains (a, b) and (a, c), then b must equal c"

Which is just a way of saying that an input of "a" cannot produce two different results.

Example: {(**2**,**4**), (**2**,**5**), (7,3)} is **not** a function because {2,4} and {2,5} means that 2 could be related to 4 **or** 5.

In other words it is not a function because it is **not single valued**



A Benefit of Ordered Pairs

We can graph them...

... because they are also coordinates!

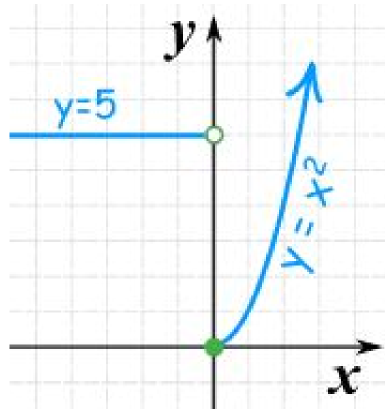
So a set of coordinates is also a function (if they follow the rules above, that is)

A Function Can be in Pieces

You can create functions that behave differently depending on the input value

Example: A function with two pieces:

- when x is less than 0, it gives 5,
- when x is 0 or more it gives x^2



Here are some example values:

x	y
-3	5
-1	5
0	0
2	4
4	16
...	...

Read more at Piecewise Functions.

Explicit vs Implicit

Before I finish, I would like to mention the terms "explicit" and "implicit".

"Explicit" is when the function shows you how to go directly from x to y, such as:

$$y = x^3 - 3$$

When you know x, you can find y

That is the classic $y = f(x)$ style.

"**Implicit**" is when it is **not** given directly such as:

$$x^2 - 3xy + y^3 = 0$$

When you know x, how do you find y?

It may be hard (or impossible!) to go directly from x to y.

"Implicit" comes from "implied", in other words shown **indirectly**.

Graphing

- The Function Grapher can only handle explicit functions,
- The Equation Grapher can handle both types (but takes a little longer, and sometimes gets it

wrong).

Conclusion

- a function **relates** inputs to outputs
- a function takes elements from a set (the **domain**) and relates them to elements in a set (the **codomain**).
- all the outputs (the actual values related to) are together called the **range**
- a function is a **special** type of relation where:
    - **every element** in the domain is included, and
    - any input produces **only one output** (not this **or** that)
- an input and its matching output are together called an **ordered pair**
- so a function can also be seen as a **set of ordered pairs**

Question 1 Question 2 Question 3 Question 4 Question 5
Question 6 Question 7 Question 8 Question 9

Search :: Index :: About :: Contact :: Contribute :: Cite This Page :: Privacy