CMPSCI 250: Introduction to Computation

Lecture #9: Properties of Functions and Relations
David Mix Barrington
12 February 2014

Relations and Functions

- Defining Functions With Quantifiers
- Total and Well-Defined Relations
- One-to-One and Onto Functions
- Bijections
- Composition and Inverse Functions
- Properties of Binary Relations on a Set
- Examples of Binary Relations on a Set

Relations and Direct Products

- Recall that when A and B are two sets, a **relation** from A to B is any set of ordered pairs, where the first element of each pair is from A and the second is from B.
- We say that the relation R is a subset of the **direct product** $A \times B$, which is the set of all such ordered pairs.

Functions

- A **function** in ordinary computing usage is an entity that gives an **output** of a given type (the **range**) whenever it is called with an input of a given type (the **domain**).
- A function **from A to B** takes input from A and gives output from B.
- A relation from A to B may or may not define a function from A to B.

Relations and Functions

- We say that the relation is a function if for each input, there is *exactly one* possible output.
- That is, for every element x of A, there is exactly one element y of B such that the pair (x, y) is in the relation.
- We can put this definition into formal terms using predicates and quantifiers.

When a Relation is a Function

- Let R be a relation from A to B. We'll write "(x, y) ∈ R" as "R(x, y)", identifying the relation with its corresponding predicate. What does it mean for R to be a function?
- Part of the answer is that each x must have at least one y such that R(x, y) is true. In symbols, we say $\forall x: \exists y: R(x, y)$. This property of a relation is called being **total**.

When a Relation is a Function

- We also require that each x may have at most one y such that R(x, y) is true -- this is the property of being **well-defined**.
- We can write that no x has more than one y, by saying $\forall x: \forall y: \forall z: (R(x, y) \wedge R(x, z)) \rightarrow (y = z)$. Another way to say this is $\neg\exists x: \exists y: \exists z: R(x, y) \wedge R(x, z) \wedge (y \neq z)$.
- A relation that is well-defined, but not necessarily total, is called a **partial function**. A non-void Java method computes a partial function since it may not terminate for all possible inputs.

Clicker Question #1

- Let **N** be the set of natural numbers {0, 1, 2, 3, ...}. Here are four binary relations on **N**. Which one is a *function* from **N** to **N**? Remember that a function must be both total and well-defined.
- (a) $A(x, y) = \{(x, y): x^2 + y^2 = 25\}$
- (b) $B(x, y) = \{(x, y): x^2 = y\}$
- (c) $C(x, y) = \{(x, y): x = 2y + 3\}$
- (d) $D(x, y) = \{(x, y): y = 6x - 4\}$

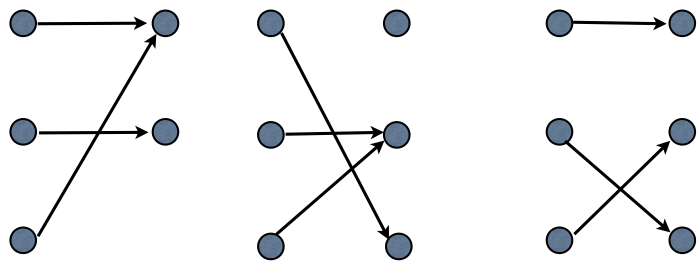
Answer #1

- Let **N** be the set of natural numbers {0, 1, 2, 3, ...}. Here are four binary relations on **N**. Which one is a *function* from **N** to **N**? Remember that a function must be both total and well-defined.
- (a) $A(x, y) = \{(x, y): x^2 + y^2 = 25\}$
- *(b) $B(x, y) = \{(x, y): x^2 = y\}$*
- (c) $C(x, y) = \{(x, y): x = 2y + 3\}$
- (d) $D(x, y) = \{(x, y): y = 6x - 4\}$

Onto Functions (Surjections)

- We can also use quantifiers to define two important properties of functions.
- A function is **onto** (also called a **surjection**) if every element of the range is the output for at least one input, in symbols $\forall y: \exists x: R(x, y)$. Note that this is not the same as the definition of "total" because the x and y are switched -- it is the **dual** property of being total.

Onto Functions (Surjections)

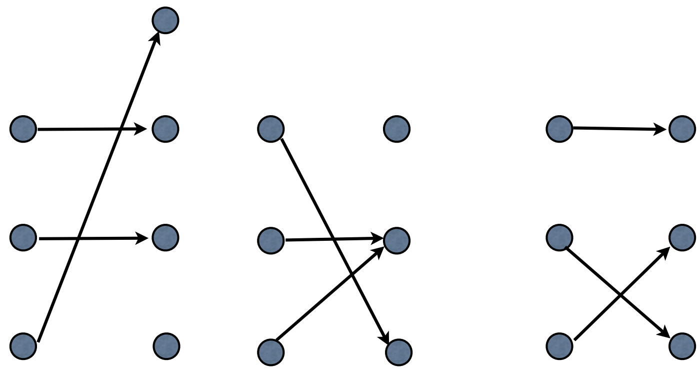


Onto, not 1-1   Not Onto   Onto and 1-1

One-to-One Functions

- A function is **one-to-one** (an **injection**) if it is never true that two different inputs map map to the same output.
- We can write this as $\forall w: \forall x: \forall y: (R(w, y) \wedge R(x, y)) \rightarrow (w = x)$, or equivalently $\neg\exists w: \exists x: \exists y: R(w, y) \wedge R(x, y) \wedge (w \neq x)$. This is obtained from the well-defined property by switching domain and range.

One-to-one Functions (Injections)



1-1, not onto

Not 1-1

Onto and 1-1

Functions and Sizes of Sets

- These properties are important in **combinatorics** -- if A and B are finite sets, we can have a surjection from A to B if and only if |A| ≥ |B|.
- Similarly, we can have an injection from A to B if and only if |A| ≤ |B|.
- (Here "|A|" denotes the number of elements in A, and "|B|" the number in B.)

Bijections

- It is possible for a function to be both onto and one-to-one. We call such a function a **bijection** (also sometimes a **one-to-one correspondence** or a **matching**).
- From what we just said about the sizes of finite sets in a surjection or injection, we can see that a bijection from A to B is possible if and only if |A| = |B|.

Clicker Question #2

- Suppose that A and B are two finite sets, and that f is a function from A to B. If f is *onto but not one-to-one*, what can we conclude about the sizes |A| and |B| of A and B?
- (a) $|A| \leq |B|$
- (b) $|A| = |B|$
- (c) $|A| > |B|$
- (d) $|A| \geq |B|$

Answer #2

- Suppose that A and B are two finite sets, and that f is a function from A to B. If f is *onto but not one-to-one*, what can we conclude about the sizes |A| and |B| of A and B?
- (a) $|A| \leq |B|$
- (b) $|A| = |B|$
- *(c) $|A| > |B|$*
- (d) $|A| \geq |B|$

Bijections

- There is an interesting theory, which we don't have time for in this course, about the sizes of **infinite** sets, where we define two sets to have the same "size" if there is a bijection from one to the other.
- A bijection from a set to itself is also called a **permutation**. The problem of sorting is to find a permutation of a set that puts it in some desired order.

Composition of Functions

- If f is a function from A to B, and g is a function from B to C, we can define a function h from A to C by the rule $h(x) = g(f(x))$. We map x by f to some element y of B, then map y by g to an element of C. This new function is called the **composition** of f and g, and is written "$g \circ f$".
- The notation $g \circ f$ is chosen so that $(g \circ f)(x) = g(f(x))$, that is, the order of f and g remains the same in these two ways of writing it.

Inverse Functions

- With quantifiers, we can define $(g \circ f)(x) = z$ to mean $\exists y: (f(x) = y) \wedge (g(y) = z)$.
- If A and C are the same set, it is possible that the function g *undoes* the function f, so that g(f(x)) is always equal to x. This can only happen when f is a bijection -- in this case A and B have the same size, and g must also be a bijection. We then say that f and g are **inverse functions** for each other.

Properties of Binary Relations

- Binary relations from a set to itself (called **relations on a set**) may or may not have certain properties that we also define with quantifiers.
- A relation R is **reflexive** if ∀x: R(x, x) is true, and **antireflexive** if ∀x: ¬R(x, x). Note that "antireflexive" is not the same thing as "not reflexive".

More Properties

- R is **symmetric** if $\forall x: \forall y: R(x, y) \rightarrow R(y, x)$, or equivalently $\forall x: \forall y: R(x, y) \leftrightarrow R(y, x)$.
- R is **antisymmetric** if $\forall x: \forall y: (R(x, y) \wedge R(y, x)) \rightarrow (x = y)$. Again "antisymmetric" is a different property from "not symmetric".
- R is **transitive** if $\forall x: \forall y: \forall z: (R(x, y) \wedge R(y, z)) \rightarrow R(x, z)$. We saw this property in the last lecture with the "smaller than" property for dogs.

Examples of Binary Relations

- The **equality relation** E is defined so that E(x, y) is true if and only if x = y.
- This relation is reflexive, symmetric, and transitive.
- We'll soon see that any relation with these three properties, called an **equivalence relation**, acts in many ways like equality.

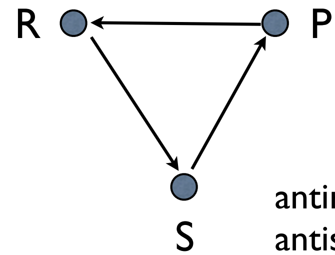
Examples of Binary Relations

- On numbers, for example, we can define LE(x, y) to mean $x \leq y$, and LT(x, y) to mean $x < y$.
- LE is reflexive, antisymmetric, and transitive, and relations with those three properties are called **partial orders**.
- LT, on the other hand, is antireflexive, antisymmetric, and transitive.

Examples of Binary Relations

- In the game of rock-paper-scissors, we can define a “beats” relation so that B(x, y) means “x beats y in the game”.
- So B(r, s), B(s, p), and B(p, r) are true and the other six possible atomic statements are false.
- This relation is antireflexive, antisymmetric, and *not* transitive.

Rock-Paper-Scissors



antireflexive -- no loops
antisymmetric -- no two-way arrows
not transitive -- two-step paths with no shortcuts

Clicker Question #3

- Let the binary relation R on **N** be defined so that R(x, y) is {(x, y): x + 3 < y}. This relation is:
- (a) both reflexive and transitive
- (b) reflexive but not transitive
- (c) transitive but not reflexive
- (d) neither reflexive nor transitive

Answer #3

- Let the binary relation R on **N** be defined so that R(x, y) is {(x, y): $x + 3 < y$}. This relation is:
- (a) both reflexive and transitive
- (b) reflexive but not transitive
- *(c) transitive but not reflexive*
- (d) neither reflexive nor transitive