Formal Foundations: A Basic Review of Sets and Functions [1]

1. Naïve Set Theory

1.1 Basic Properties of Sets

A *set* is a group of objects. Any group of objects a, b, c forms a *set*.

(1) **Representation of Sets 1**
The most basic way to represent a set is to enclose a list of its members in curly brackets:
{ a, b, c } = the set consisting of a, b, and c

The things constituting a set are its *members* or its *elements*.

(2) **Representing Set Membership**
- If an object *b* is a member/element of a set *A*, then we write $b \in A$.
- If an object *b* is not a member/element of a set *A*, then we write $b \notin A$.

Anything can be a member of a set, including another set.

(3) **Some Possible Sets**
- { a, b, c, { d, e, f } } = the set consisting of a, b, c, and the set consisting of d, e, f
- { { f, e } } = the set whose elements are the set whose elements are f, e

Sets are entirely defined by who their members are. Thus, order doesn't matter in the representation of a set (4a). Also, repeating the name of an element doesn't matter in the representation of a set (4b).

(4) **Some Equivalences**
a. { a, b, c } = { a , c, b } = { b, a, c } = { b, c, a } = { c, a, b } = { c, b, a }
b. { a, a, b, b, c, c, c } = { a, b, c }

A set can consist of only one entity. Such a set is called a *singleton.*

(5) **Examples of Singleton Sets**
- { a } = The singleton of a
- { { f, e } } = The singleton of the set consisting of f and e

There is exactly one, unique set that doesn't have any members of at all. We call this *the empty set*, or *the null set*.

(6) **Representation of the Empty/Null Set**
The empty/null set can be written as '{ }'. Another common notation for it is '∅'.

[1] These notes summarize the review of sets and functions provided by Heim & Kratzer (1998: 3-11). For further review, see the optional readings: Chierchia & McConnell-Ginet (2000: 529-540), and Partee *et al.* (1990: 3-36).

1.2 Relations Between Sets

If two sets share some members in common, then they are said to *overlap*. If two sets *don't* share any members in common, they are said to be *disjoint*.

(6) **Some Overlapping Sets**
- { a, b, c } and { c, d, e }

(7) **Some Disjoint Sets**
- { a, b, c } and { d, e, f }
- { a, b, c } and { { b, c, }, d }

If every member of a set *A* is also a member of a set *B*, then *A* is a *subset* of *B*, and *B* is a *superset* of *A*.

(8) **Some Subsets and Supersets**
- { a, b, c } is a *subset* of { a, b, c, d }
- { x, y, z } is a *superset* of { x, y }

(9) **Representation of Subset and Superset Relations**
- If *A* is a subset of *B*, then we write '$A \subseteq B$'.
- If *B* is a superset of *A*, then we write '$B \supseteq A$'.

(10) **Important Consequence**
For every set *A*, $A \subseteq A$, and $A \supseteq A$.

(11) **Proper Subsets and Supersets**
- If $A \subseteq B$ and $A \neq B$, then *A* is a *proper subset* of *B*, and we write '$A \subset B$'
- If $A \supseteq B$ and $A \neq B$, then *A* is a *proper superset* of *B*, and we write '$A \supset B$'

1.3 Operations on Sets

Given two sets *A* and *B*, the *union* of *A* and *B* is the smallest set that contains all the members of A *and* all the members of *B*.

(12) **Illustration of Set Union** The union of { a, b, c } and { c, d, e } is { a, b, c, d, e }

(13) **Representation of Set Union** For 'the union of *A* and *B*', we may write '$A \cup$ B'

Given two sets *A* and *B*, the *intersection* of *A* and *B* is the largest set containing only elements that *A* and *B* share in common.

(14) **Illustration of Set Intersection** The intersection of { a, b, c } and { c, d, e } is { c }

(15) **Representation of Set Intersection** For 'the intersection of *A* and *B*', we write '$A \cap B$'

1.4 Defining Sets

Sets can have a very large – even infinite – number of members. Clearly, in such cases, we cannot define a set by listing its members inside curly brackets. Here are some ways that we can.

(16) **Method 1: Using Names**

As we have already been doing, we can use capital letters to refer to particular sets, as long as it is clear what sets those letters stand for.

- *N* = the set of all natural numbers
- *S* = the set of all students at UMass

(17) **Method 2: Abstraction Notation**

We can define sets using a special notation referred to as 'set abstraction'.

$\{ x : \varphi(x) \}$ = 'the set of entities x such that $\varphi(x)$'

(18) **Illustration of Set Abstraction**

- **{ x : x is a student at UMass }** =
  the set of entities x such that x is a student at UMass =
  the set of students at UMass

- **{ y : my father once dated a girl who once put y in her hair }** =
  the set of entities y such that my dad dated a girl who once put y in her hair.

- **{ z : { s : z hates s } = ∅ }** =
  the set of entities z such that the set of entities s such that z hates s is null =
  the set of entities that don't hate anything (or anyone)

(19) **A General Rule of Thumb for Set Abstraction**

How do we determine whether $a \in \{ x : \varphi(x) \}$?

- Substitute *a* for the variable 'x' in '$\varphi(x)$', and ask whether '$\varphi(a)$' is *true*.
  If it is, then $a \in \{ x : \varphi(x) \}$
  If it isn't, then $a \notin \{ x : \varphi(x) \}$

(20) **Some Comprehension Exercises**

a. Is the following true:
Barack Obama $\in \{ x : \{ y : y \text{ is a parent of } x \} \cap \{ z : z \text{ is American} \} = \emptyset \}$

b. Is the following true: $5 \in \{ x : x + 1 \in \{ y : 5 \in \{ z : z < y \} \} \}$

2. Basic Function Theory

2.1 Ordered Pairs

For any two entities x and y, there exists the *ordered pair* <x, y>.

Unlike sets, ordered pairs are partly defined by the order that their elements appear in (hence the name). Therefore, the following inequivalence generally holds:

(21) **General Property of Ordered Pairs**
If $x \neq y$, then $\langle x, y \rangle \neq \langle y, x \rangle$

2.2 Basic Properties of Functions

Just as we can have sets of entities and sets of sets, we can also have sets of ordered pairs.

(22) **Set of Ordered Pairs**
{ <a, b>, <a, d>, <d, e> }

There is a specific *kind* of set of ordered pairs that is of crucial importance to mathematics, logic and natural language semantics: *the function*.

(23) **Function**
A *function* is any set of ordered pairs *f* that satisfies the following property:

- For any x, if there are y and z such that $\langle x,y \rangle \in f$ and $\langle x,z \rangle \in f$, then $y = z$
  (In other words, if $\langle x,y \rangle \in f$, then there is no z such that $z \neq y$ and $\langle x,z \rangle \in f$)
  (In other words, there are never two ordered pairs p_1 and p_2 in *f* where p_1 and p_2 have the same first member, but different second members)

(24) **Illustration of Functions**

- The following set is a function: { <1,1>, <2,4>, <3,9> }
- The following set is *not* a function: { <1,1>, <2,4>, <2,9> }

Why are functions important? Well, because of the definition in (23), we know that whenever *f* is a function, the following notation makes sense.

(25) **Notation: '*f*(x)'**
'*f*(x)' $=_{def}$ the unique y such that $\langle x,y \rangle \in f$

(26) **Quick Comprehension Question**
If *f* = { <1,1>, <2,4>, <3,9> }, which of the following are true:
a. f(2) = 4
b. f(1) = 1
c. f(9) = 3

(27) **A Key Analogy: Functions as 'Machines'**

We can think of functions as little deterministic machines.

- The ordered pairs <x,y> in the function are pairs of *input* and *output*.
- The function takes the first member of the ordered pair as *input*, and (thanks to the definition in (23)), spits out the second member of the pair as *output.*

(28) **Informal Reading of the Notation '*f*(x)'**

For the equation '*f*(x) = y':
- We can refer to '*f*' as 'the function'
- We can refer to 'x' as 'the argument'
- We can refer to 'y' as 'the value'

Thus, we can informally 'read' the term '*f*(x)' as:
- "The function *f* applied to (the argument) x."
- "The function *f* taking x as argument."
- "*f* of x"

And, we can informally 'read' the equation '*f*(x) = y' as:
- "The function *f* maps/takes (the argument) x to (the value) y."

Two key properties of a function are its *domain* and its *range*.

(29) **Domain of a Function**
For any function *f*, the *domain* of *f* is the set of all those entities x such that there is some y such that $<x,y> \in f$.
- That is, it's the set of all the possible arguments (inputs) of *f*.

(30) **Range of a Function**
For any function *f*, the *range* of *f* is the set of all those entities y such that there is some x such that $<x,y> \in f$.
- That is, it's the set of all the possible values (outputs) of *f*.

We say that a given function *maps* its domain *onto* its range.

For any function *f*, if B is a superset of the range of *f*, then we say that *f* maps its domain *into B*.

(31) **Notation for Domains and Ranges**
If the function *f* maps the set *A into* the set *B*, then we may write '$f: A \rightarrow B$'

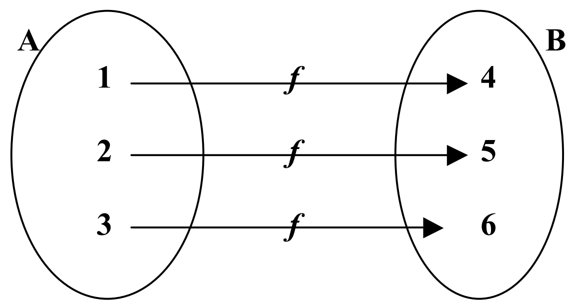
- Note: if we write '$f: A \rightarrow B$', that means that *B* is a *superset* of the range of *f*. (consequently, *B* is not necessarily the range itself…)

(32) **Definedness and Undefinedness**

- If x is not in the domain of the function *f*, then we say that *f*(x) is **undefined**.
- Equivalently, x is **undefined** for the function *f* if there is no y such that <x,y> ∈ *f*.

(33) **Visual Metaphor for Functions**

f = { <1,4>, <2,5>, <3,6> }



2.3 Defining Functions

Functions can be very large, and often consist of an infinite number of ordered pairs. Thus, in the general case, we cannot define functions simply be listing their members in set notation.

(34) **Notation for Functions**
We can define a function by specifying a condition that must be met by every ordered pair in the function.

f : A → B
for every x ∈ A, f(x) = φ(x)

"The function f that (i) maps A into B, and (ii) for every argument x, maps x to φ(x)"

(35) **Illustration of the Notation**

f : *N* → *N*
for every x ∈ *N*, f(x) = $x^2 + 2$

"The function f that (i) maps the natural numbers into the natural numbers, and (ii) for every natural number x, maps x to x^2 +2"

h : { x : x is a person } → { x : x is a person }
for every x ∈ { x : x is a person }, h(x) = the father of x

"the function h that (i) maps the set of people into the set of people, and (ii) for every person x, maps x to the father of x."

(36) **Some Quick Comprehension Questions**

a. Consider the following function: { <1,1>, <2,4>, <3,9> }. How can we represent this using the notation in (34)?

b. Consider the function *f* in (35). Let's try writing out part of this function as a set of ordered pairs.

If a function happens to be finite in size, then another 'snazzy' way to represent it is via the following 'table notation'. (This is used occasionally in the early chapters of Heim & Kratzer.)

(37) **Table Notation for (Finite) Functions**

$$f = \left(\begin{array}{l} \text{argument 1} \rightarrow \text{value 1} \\ \text{argument 2} \rightarrow \text{value 2} \\ \ldots \\ \text{argument n} \rightarrow \text{value n} \end{array}\right)$$