Lectures 1-3. Basic Concepts of Set Theory, Functions and Relations; and Trees

Reading: Chapter 1 – Chapter 3 of Partee, ter Meulen, and Wall (PtMW), pp.3—53.

0. Goals

We presuppose that you are probably familiar with most of the notions and constructions we will discuss in Section 1 (roughly lecture 1). So our main goals here are to remind you of the main notions and definitions, establish some background for the later lectures, and discuss the use of these notions in linguistics. Some of the notions in Sections 2 and 3 may be new.

1. Basic Concepts of Set Theory.

1.1. Sets and elements

Set theory is a basis of modern mathematics, and notions of set theory are used in all formal descriptions. The notion of set is taken as "undefined", "primitive", or "basic", so we don't try to *define* what a set is, but we can give an informal description, describe important properties of sets, and give examples. All other notions of mathematics can be built up based on the notion of set.

Similar (but informal) words: collection, group, aggregate.

Description and terminology: a *set* is a collection of objects (entities) which are called the *members* or *elements* of that set. If we have a set we say that some objects *belong* (or *do not belong*) to this set, *are* (or *are not*) *in* the set. We say also that sets *consist* of their elements.

Examples: the set of students in this room; the English alphabet may be viewed as the set of letters of the English language; the set of even numbers; etc.

So sets can consist of elements of various natures: people, physical objects, numbers, signs, other sets, etc. (We will use the words *object* or *entity* in a very broad way to include all these different kinds of things.)

The membership criterion for a set must in principle be well-defined, and not vague. If we have a set and an object, it is possible that we do not know whether this object belongs to the set or not, because of our lack of information or knowledge. (E.g. "The set of people in this room over the age of 28", if we don't know everyone's age.) But the answer should exist, at any rate in principle. It could be unknown, but it should not be vague. If the answer is vague for some putative set-description, we cannot consider that a real description of a set. Another thing: If we have a set, then for every two elements of it, *x* and *y*, it should not be vague whether $x = y$, or they are different.

Sometimes we simply assume for the sake of examples that a description is not vague when perhaps for other purposes it would be vague – e.g., the set of all red objects.

Possible examples of non-sets: "the set of all words of a given language (Russian, English, etc)" [We will call it a non-set if we don't believe that the class is really well-defined]; the set of faces I saw in the street [similarly], the set of all sets. [This last one deserves separate discussion; see discussion of Russell's paradox on pp. 7-8] Because we often just implicitly "stipulate" that our predicates are unambiguously clearcut and not vague, it is hard to give clear examples of descriptions that definitely do not pick out sets.

Sets can be *finite* or *infinite*. (We'll discuss infinity more later.)
There is exactly one set, the *empty set*, which has no members at all.
A set with only one member is called a *singleton* or a *singleton set*.

Notation: *A, B, C, ...* for sets; *a, b, c, ...* or *x, y, z, ...* for members.
$b \in A$ if *b* belongs to *A* ($B \in A$ if both *A* and *B* are sets and *B* is a member of *A*)
and $c \notin A$, if *c* doesn't belong to *A*.
$\varnothing$ is used for the empty set.

1.2. Specification of sets

There are three main ways to specify a set:
(1) by listing all its members (*list notation*);
(2) by stating a property of its elements (*predicate notation*);
(3) by defining a set of rules which generates (defines) its members (*recursive rules*).

List notation. The first way of course is suitable only for finite sets. In this case we list names of elements of a set, separate them by commas and enclose them in braces:
Examples: {1, 12, 45}, {George Washington, Bill Clinton}, {a,b,d,m}.

Note that we do not care about the order of elements of the list, and elements can be listed several times. {1, 12, 45}, {12, 1, 45,1} and {45,12, 45,1} are different representations of the same set (see below the notion of identity of sets).

Predicate notation. Example:
$\{x \mid x$ is an even number greater than 3.$\}$
Read: "the set of all x such that x is an even number greater than 3"
So the second part of this notation is a property the members of the set share (a condition or a predicate which holds for members of this set).

Other examples:
$\{x \mid x$ is a letter of Russian alphabet$\}$
$\{y \mid y$ is a student of UMass and y is older than 25 $\}$

General form:
$\{x \mid P(x)\}$, where P is some predicate (condition, property).

The language to describe these predicates is not usually fixed in a strict way. But it is known that unrestricted language can result in *paradoxes*. Example: $\{x \mid x \notin x\}$.
(Russell's paradox: see PtMW pp 7-8. Type theory is one way to guarantee avoidance of this kind of paradox: require sets always to be of higher type than their members.)

Recursive rules. (Always safe.) Example – the set E of even numbers greater than 3:
a) $4 \in E$
b) if $x \in E$, then $x + 2 \in E$
c) nothing else belongs to E.

The first rule is the basis of recursion, the second one generates new elements from the elements defined before and the third rule restricts the defined set to the elements generated by rules a and b. (The third rule should always be there; sometimes in practice it is left implicit.)

1.3. Identity and cardinality

Two sets are identical if and only if they have exactly the same members. So $A = B$ iff for every x, $x \in A \Leftrightarrow x \in B$.

For example, $\{0,2,4\} = \{x \mid x$ is an even natural number less than 5$\}$

From the definition of identity it follows that there exists only one empty set; its identity is fully determined by its absence of members. Note that empty list notation {} is not usually used for the empty set, we have a special symbol $\varnothing$ for it.

The number of elements in a set A is called the *cardinality* of A, written $|A|$. The cardinality of a finite set is a natural number. Infinite sets also have cardinalities but they are not natural numbers. We will not discuss cardinalities of infinite sets yet.

1.4. Subsets

A set A is a *subset of a set B* iff every element of A is also an element of B. Such a relation between sets is denoted by $A \subseteq B$. If $A \subseteq B$ and $A \neq B$ we call A a *proper subset* of B and write $A \subset B$. (Caution: sometimes $\subset$ is used the way we are using $\subseteq$.)
        Both signs can be negated using the slash / through the sign.
Examples:
$\{a,b\} \subseteq \{d,a,b,e\}$ and $\{a,b\} \subset \{d,a,b,e\}$, $\{a,b\} \subseteq \{a,b\}$, but $\{a,b\} \not\subset \{a,b\}$.

Note that the empty set is a subset of every set. $\varnothing \subseteq A$ for every set A. Why?

1.5. Power sets

The set of all subsets of a set A is called the *power set* of A and denoted as $\wp(A)$ or sometimes as 2^A.
For example, if $A = \{a,b\}$, $\wp(A) = \{\varnothing, \{a\}, \{b\}, \{a,b\}\}$.

From the example above: a $\in$ A; $\{a\} \subseteq$ A; $\{a\} \in \wp(A)$
        $\varnothing \subseteq$ A; $\varnothing \notin$ A; $\varnothing \in \wp(A)$; $\varnothing \subseteq \wp(A)$

1.6 and 1.7. Operations on sets: union, intersection, difference, complement

We define several operations on sets. Let A and B be arbitrary sets.

        The *union* of A and B, written $A \cup B$, is the set whose elements are just the elements of A or B or of both. In the predicate notation the definition is
        $A \cup B =_{def} \{ x \mid x \in A \text{ or } x \in B\}$

Examples. Let $K = \{a,b\}$, $L = \{c,d\}$ and $M = \{b,d\}$, then

        $K \cup L$ = $\{a,b,c,d\}$
        $K \cup M$ = $\{a,b,d\}$
        $L \cup M$ = $\{b,c,d\}$
        $(K \cup L) \cup M = K \cup (L \cup M)$ = $\{a,b,c,d\}$
        $K \cup K$ = K
        $K \cup \varnothing = \varnothing \cup K = K$ = $\{a,b\}$.

        There is a nice method for visually representing sets and set-theoretic operations, called *Venn diagrams*. Each set is drawn as a circle and its members represented by points within it. The diagrams for two arbitrarily chosen sets are represented as partially intersecting – the most general case – as in Figure 1–1 below. The region designated '1' contains elements which are members of A but not of B; region 2, those members in B but

not in A; and region 3, members of both B and A. Points in region 4 outside the diagram represent elements in neither set.

The Venn diagram for the union of A and B is shown in Figure 1–2. The results of operations in this and other diagrams are shown by shading areas.

[PtMW, p.13, Fig.1-1, 1-2]

The *intersection* of A and B, written $A \cap B$, is the set whose elements are just the elements of both A and B. In the predicate notation the definition is

$$A \cap B =_{\text{def}} \{ x \mid x \in A \text{ and } x \in B \}$$

Examples:

$$
\begin{aligned}
K \cap L \quad &= \quad \varnothing \\
K \cap M \quad &= \quad \{b\} \\
L \cap M \quad &= \quad \{d\} \\
(K \cap L) \cap M &= K \cap (L \cap M) = \varnothing \\
K \cap K \; &= K \\
K \cap \varnothing &= \varnothing \cap K = \varnothing.
\end{aligned}
$$

The general case of intersection of arbitrary sets A and B is represented by the Venn diagram of Figure 1–3. The intersection of three arbitrary sets A,B and C is shown in the Venn diagram of Figure 1–4.

[PtMW, p. 14, Fig 1-3, p.15, Fig. 1-4]

Another binary operation on arbitrary sets is the *difference*, written $A - B$, which 'subtracts' from A all elements which are in B. The predicate notation defines this operation as follows:

$$A - B =_{\text{def}} \{ x \mid x \in A \text{ and } x \notin B \}$$

Examples:

$$K - L \quad = \quad \{a,b\}$$
$$K - M \quad = \quad \{a\}$$
$$L - M \quad = \quad \{c\}$$
$$K - K = \varnothing$$
$$K - \varnothing = K$$
$$\varnothing - K = \varnothing.$$

The Venn diagram for the set-theoretic difference is shown in Figure 1–5.

$A - B$ is also called the *relative complement* of B relative to A. This operation is to be distinguished from the *complement* of a set A, written A', which is the set consisting of everything not in A. In predicate notation

A' $=_{\text{def}} \{ x \mid x \notin A \}$

It is natural to ask, where do these objects come from which do not belong to A? In this case it is presupposed that there exists a *universe of discourse* and all other sets are subsets of this set. The universe of discourse is conventionally denoted by the symbol U. Then we have

A' $=_{\text{def}} U - A$

The Venn diagram with a shaded section for the complement of A is shown in Figure1–6.

[PtMW, p.16, Fig. 1-5, p.16, Fig. 1-6]

1.8. Set-theoretic equalities

There are a number of general laws about sets which follow from the definitions of set-theoretic operations, subsets, etc. A useful selection of these is shown below. They are grouped under their traditional names. These equations below hold for any sets X, Y, Z:

1. *Idempotent Laws*
   (a) $X \cup X = X$ (b) $X \cap X = X$

2. *Commutative Laws*
   (a) $X \cup Y = Y \cup X$ (b) $X \cap Y = Y \cap X$

3. *Associative Laws*
   (a) $(X \cup Y) \cup Z = X \cup (Y \cup Z)$ (b) $(X \cap Y) \cap Z = X \cap (Y \cap Z)$

4. *Distributive Laws*
   (a) $X \cup (Y \cap Z) = (X \cup Y) \cap (X \cup Z)$ (b) $X \cap (Y \cup Z) = (X \cap Y) \cup (X \cap Z)$

5. *Identity Laws*
   (a) $X \cup \varnothing = X$ (c) $X \cap \varnothing = \varnothing$
   (b) $X \cup U = U$ (d) $X \cap U = X$

6. *Complement Laws*
   (a) $X \cup X' = U$ (c) $X \cap X' = \varnothing$
   (b) $(X')' = X$ (d) $X - Y = X \cap Y'$

7. *DeMorgan's Laws*
   (a) $(X \cup Y)' = X' \cap Y'$ (b) $(X \cap Y)' = X' \cup Y'$

8. Consistency Principle
   (a) $X \subseteq Y$ iff $X \cup Y = Y$ (b) $X \subseteq Y$ iff $X \cap Y = X$

We will see later that operations on subsets of a set form a *Boolean algebra*.

Homework 1.

1) Exercises from PtMW:

Chapter 1, pp. 23-26. ##2, 5, 6(a,g,m), 7(a,f,l), 8(b,c), 9b, 11a,c,d.

2. Relations and Functions

2.1. Ordered pairs and Cartesian products

As we see, there is no order imposed on the elements of a set. To describe functions and relations we will need the notion of an *ordered pair*, written *<a,b>*, for example, in which *a* is considered the *first member (element)* and *b* is the *second member (element)* of the pair. The ordered pair can be defined as follows:

$$<a,b> =_{def} \{\{a\}, \{a,b\}\}$$

So, in general, $<a,b> \neq <b,a>$.

If we have two sets *A* and *B*, we can form ordered pairs from them by taking an element of *A* as the first member of the pair and an element of *B* as the second member. The *Cartesian product* of *A* and *B*, written $A \times B$, is the set consisting of all such pairs. The predicate notation defines it as

$$A \times B =_{def} \{<x,y> \mid x \in A \text{ and } y \in B\}$$

Note that according to the definition, if either A or B is $\varnothing$, then $A \times B = \varnothing$.

Here are some examples of Cartesian products:

Let $K = \{a,b,c\}$ *and* $L = \{1,2\}$, then

$$
\begin{aligned}
K \times L &= \{<a,1>,<a,2>,<b,1>,<b,2>,<c,1>,<c,2>\} \\
L \times K &= \{<1,a>,<2,a>,<1,b>,<2,b>,<1,c>,<2,c>\} \\
L \times L &= \{<1,1>,<1,2>,<2,1>,<2,2>\}
\end{aligned}
$$

The definition of ordered pairs can be extended to ordered triples and in general to ordered *n*-tuples for any natural *n*. For example, ordered triples are defined as

$$<a,b,c> =_{def} <<a,b>,c>$$

And for three sets *A*, *B* and *C* the Cartesian product can be defined as
$$A \times B \times C =_{def} ((A \times B) \times C)$$

In the case when $A = B = C = \dots$ a special notation is used: $A \times A = A^2$, $A \times A \times A = A^3$, etc. And we put $A^1 =_{def} A$.

2.2. Relations

In natural language *relations* are a kind of links existing between objects. Examples: 'mother of', 'neighbor of', "part of" etc. These are binary relations. Formally we will define relations between elements of sets.

If *A* and *B* are any sets and $R \subseteq A \times B$, we call *R* a *binary relation from A to B* or a *binary relation between A and B*. A relation $R \subseteq A \times A$ is called a relation *in* or *on A*.

The set **dom** $R = \{a \mid <a,b> \in R$ for some $b\}$ is called the *domain* of the relation R and the set **range** $R = \{b \mid <a,b> \in R$ for some $a\}$ is called the *range* of the relation R.

We may visually represent a relation R between two sets A and B by arrows in a diagram displaying the members of both sets. In Figure 2-1, $A = \{a.b\}$, $B = \{c,d,e\}$ and the arrows represent a set-theoretic relation $R = \{<a,d>,<a,e>, <b,c>\}$.

[PtMW, p. 29, Fig.2-1]

Let us consider some operations on relations. The *complement* of a relation $R \subseteq A \times B$ is defined as

$$R' =_{\text{def}} (A \times B) - R$$

The *inverse* of a relation $R \subseteq A \times B$ is defined as the relation $R^{-1} \subseteq B \times A$, $R^{-1} =_{\text{def}} \{<b,a> \mid <a,b> \in R\}$. Note that $(R^{-1})^{-1} = R$.

For example, for the relation R represented in Figure 2-1
$R' = \{<a,c>,<b,d>,<b,e>\}$ and $R^{-1} = \{<d,a>,<e,a>,<c,b>\}$.

We have focused so far on *binary* relations, i.e., sets of ordered pairs. In a similar way we could define *ternary*, *quaternary* or just *n-place* relations consisting respectively of ordered triples, quadruples or n-tuples. A unary relation R on a set A is just a subset of the set A.

2.3. Functions

A function is generally represented in set-theoretic terms as a special kind of relation. A relation *F* from *A* to *B* is called a function from *A* to *B* if and only if it meets both of the following conditions:

1. Each element in the domain of *F* is paired with just one element in the range, i.e., from $<a,b> \in F$ and $<a,c> \in F$ follows that $b = c$.

2. The domain of *F* is equal to *A*, **dom***F* = *A*.

For example, consider the sets $A = \{a.b\}$ and $B = \{1,2,3\}$. The following relations from *A* to *B* are functions from *A* to *B*:

$P = \{<a,1>,<b,1>\}$
$Q = \{<a,2>,<b,3>\}$
The following relations from *A* to *B* are not functions from *A* to *B*:

$S = \{<a,1>\}$
$T = \{<a,2>,<b,1>,<b,3>\}$

S does not satisfy the condition 2, and *T* fails to meet condition 1. *S* is a function on the smaller domain $\{a\}$; *T* is not a function at all.

Much of the terminology used in talking about functions is the same as that for relations. We say that a function with domain *A* and range a subset of *B* is a function *from A to B*, while one in $A \times A$ is said to be a function *in* or *on A*. The notation '*F*: $A \rightarrow B$' is used for '*F* is a function from *A* to *B*'. Elements of the domain of a function are called *arguments* and their correspondents in the range, *values*. If $<a,b> \in F$, the familiar notation $F(a) = b$ is used. 'Map', 'mapping' are commonly used synonyms for 'function'. A function *F*: $A^n \rightarrow A$ is also called an *n*-ary *operation* in *A*.

Functions as processes. Sometimes functions are considered in a different way, as processes, something like devices or boxes with inputs and outputs. We put the argument in the input and get the value of the function in output. In this case the set of ordered pairs in our definition is called the *graph* of the function.
Sometimes *partial* functions are considered. In this case the condition 2 in our definition can fail.

Some terminology. Functions from *A* to *B* in the general case are said to be *into B*. If the range of the function equals *B*, then the function is *onto B* (or *surjection*). A function *F*: $A \rightarrow B$ is called *one-to-one* function (or *injection*) just in case no member of *B* is assigned to more than one member of *A* (so if $a \neq b$, then $F(a) \neq F(b)$). A function which is both one-to-one and onto is called a *one-to-one correspondence* (or *bijection*). It is easy to see that if a function *F* is one-to-one correspondence, then the relation F^{-1} is a function and one-to-one correspondence.

In Figure 2-2 three functions are indicated by the same sort of diagrams we introduced previously for relations. It is easy to see that functions *F* and *G* are onto but *H* is not.

[PtMW, p. 32, Fig.2-2]

One useful class of functions are *characteristic functions* of sets. The *characteristic function of a set S,* considered as a subset of some larger domain *D,* is defined as follows:

$$F_S : D \rightarrow \{0,1\} : F_S(x) = 1 \text{ iff } x \in S$$
$$F_S(x) = 0 \text{ otherwise}$$

There is a one-to-one correspondence between sets and their characteristic functions. In semantics, where it is common to follow Frege in viewing much of semantic composition as carried out by function-argument application, it is often convenient to work with the characteristic functions of sets rather than with sets directly. Characteristic functions are used in many other applications as well.

2.4. Compositions

Given two functions $F: A \rightarrow B$ and $G: B \rightarrow C$, we may form a new function from *A* to *C*, called the *composition* of *F* and *G,* written $G \circ F$. Function composition is defined as

$$G \circ F =_{\text{def}} \{<x,z> \mid \text{for some } y,<x,y> \in F \text{ and } <y,z> \in G \}$$

Figure 2-3 shows two functions *F* and *G* and their composition.

[PtMW, p. 33, Fig.2-3]

The function $F: A \rightarrow A$ such that $F = \{<x,x> \mid x \in A\}$ is called the *identity function on A*, written id_A (or 1_A). Given a function $F: A \rightarrow B$ that is a one-to-one correspondence, we have the following equations: $F^{-1} \circ F = id_A$, $F \circ F^{-1} = id_B$.

The definition of composition need not be restricted to functions but can be applied to relations in general. Given relations $R \subseteq A \times B$ and $S \subseteq B \times C$ the composite of R and S, written $S \circ R =_{\text{def}} \{<x,z> \mid$ for some y, $<x,y> \in R$ and $<y,z> \in S \}$

Homework 2.
Chapter 2, pp 36. ## 1(a)(i, iv, vi), (b)(ii, iii, v), (c); 2; 3; 4.

3. Properties of Relations

3.1. Reflexivity, symmetry, transitivity, and connectedness
We consider here certain properties of binary relations. All these properties apply only to relations *in* (*on*) a (single) set, i.e., in $A \times A$ for example.

Reflexivity. Given a set A and a relation R in A, R is *reflexive* iff all the ordered pairs of the form $<x,x>$ are in R for every x in A. A relation which fails to be reflexive is called *nonreflexive*, but if it contains no ordered pair $<x,x>$, it said to be *irreflexive*.

Another way to state the definitions above is to use the identity relation id_A. Relation R is reflexive iff $id_A \subseteq R$, it is nonreflexive iff $id_A \not\subseteq R$, and it is irreflexive iff $id_A \cap R = \varnothing$.

Examples. Relations "= " and "≥" on the set **N** of natural numbers and relations "⊇" and "⊆" between sets are reflexive. Relations "≠" and "<" on **N** are nonreflexive and irreflexive.

Remember that we always consider relations <u>in some set</u>. And a relation (considered as a set of ordered pairs) can have different properties in different sets. For example, the relation $R = \{<1,1>, <2,2>\}$ is reflexive in the set $A_1 = \{1,2\}$ and nonreflexive in $A_2 = \{1,2,3\}$ since it lacks the pair $<3,3>$ (and of course it nonreflexive in **N**).

Symmetry. Given a set A and a relation R in A, R is *symmetric* iff for every ordered pair $<x,y>$, if $<x,y>$ is in R, then the pair $<y,x>$ is also in R. If for some $<x,y>$ in R, the pair $<y,x>$ is not in R, then R is *nonsymmetric*. If it is never the case that for any $<x,y>$ in R, the pair $<y,x>$ is in R then the relation is called *asymmetric*. Note that an asymmetric relation must be irreflexive. [Why?] A relation R is *anti-symmetric* if whenever both $<x,y>$ and $<y,x>$ are in R, then $x = y$. Note that a relation need not be reflexive to be anti-symmetric. [Why?]

Examples. The relation "*brother of*" is nonsymmetric in the set of all people, but it can be symmetric in some set, say, in the set A = {John, Peter, Bill}, if John and Bill are brothers. [Can you think of a set in which it is asymmetric?]

Transitivity. A relation R is *transitive* iff for all ordered pairs $<x,y>$ and $<y,z>$ in R, the pair $<x,z>$ is in R. If a relation fails to meet the definition of transitivity, it is *nontransitive*. If for no pairs $<x,y>$ and $<y,z>$ in R, the pair $<x,z>$ is in R, then the relation is *intransitive*.

Examples. Relations =, > and ≥ are transitive in the set of natural numbers. Is the relation "friend of" in the set of all people transitive?

Connectedness. A relation R in A is *connected* iff for every two *distinct* elements x and y in A, $<x,y> \in R$ or $<y,x> \in R$ (or both).

3.2. Diagrams of relations

It may be helpful to demonstrate the properties of relations representing them in relational diagrams. The members of the relevant set are represented by labeled points. If x is related to y, i.e. $<x,y> \in R$, an arrow connects the corresponding points. For example, Figure 3-1 represents the relation

> R = {$<1,2>,<2,1>,<2,2>, <1,1>,<2,3>,<3,3>$}

> [PtMW, p. 43, Fig.3-1]

It is apparent from the diagram that the relation is reflexive, since every point bears a loop. The relation is non-symmetric since there is no arrow from 3 to 2 (but there is one from 2 to 3). It cannot be called asymmetric or antisymmetric, since 1 is related to 2 and 2 is related to 1. It is not transitive since 1 is related to 2 and 2 to 3, but there is no arrow from 1 to 3.

3.3. Classes of relations

Using properties of relations we can consider some important classes of relations.

3.3.1. Equivalence relation.

 An equivalence relation is a relation which is reflexive, symmetric and transitive. For every equivalence relation there is a natural way to divide the set on which it is defined into mutually exclusive (disjoint) subsets which are called *equivalence classes*. We write $[[x]]$ for the set of all y such that $<x,y> \in R$. Thus, when R is an equivalence relation, $[[x]]$ is the equivalence class which contains x.

The set $A/R =_{\text{def}} \{[[x]] \mid x \in A\}$ is called a *quotient set* of the set A by the equivalence R. A/R is a subset of $\wp(A)$. For every equivalence relation R, the function **nat**(R): $A \rightarrow A/R$ mapping every element $x \in A$ onto $[[x]]$ is called a *natural mapping* of A onto A/R.

 Examples. The relations "has the same hair color as" or "is the same age as" in the set of people are equivalence relations. The equivalence classes under the relation "has the same hair color as" are the set of blond people, the set of red-haired people, etc.
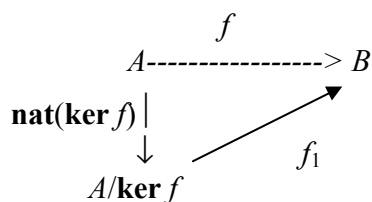
Partitions. Given a non-empty set A, a *partition* of A is a collection of non-empty subsets of A such that (1) for any two distinct subsets X and Y, $X \cap Y = \varnothing$ and (2) the union of all the subsets in collection equals A. The subsets of A that are members of a partition of A are called *cells* of that partition.

There is a close correspondence between partitions and equivalence relations. Given a partition of set A, the relation $R = \{<x,y> \mid x$ and y are in the same cell of the partition of $A\}$ is an equivalence relation in A. Conversely, given an equivalence relation R in A, there exists a partition of A in which x and y are in the same cell iff $<x,y> \in R..$

Functions and equivalences. Every function $f: A \rightarrow B$ determines an equivalence relation $f^{-1} \circ f$ on the set A. This equivalence relation is called the *kernel* of f, written **ker** f; so **ker** $f =_{\text{def}} f^{-1} \circ f$. The equivalence relation we are defining is one on which x is equivalent to y if $f(\text{x}) = f(\text{y})$. Note that f^{-1} is not necessarily a function. We are dealing here with composition of *relations*. (Composition of functions is a special case of composition of relations.)

For the quotient set of this equivalence we have the one-to-one mapping f_1: $A/\textbf{ker}\,f \rightarrow B$ such that $f_1([[x]]) = f(x)$. We can show that the following equation holds: $f = f_1 \circ \textbf{nat}\,(\textbf{ker}\,f)$.

This equation can be pictured in the form of a *commutative diagram*:

$$
\begin{array}{ccc}
 & f & \\
A\text{-----------------}\!\!> & & B \\
\textbf{nat}(\textbf{ker}\,f) \downarrow & \nearrow & \\
A/\textbf{ker}\,f & f_1 &
\end{array}
$$

Example. Consider two sets *Countries* = {Germany, England, India, China}, *Continents* = {Europe, Asia, Australia, North America} and a function f: *Countries* $\rightarrow$ *Continents* mapping every country to its continent. Then
The equivalence relation **ker** f = {<Germany, Germany>, <Germany, England>, <England, England>, <England, Germany>, <India, India>, <India, China>, <China, China>, <China, India>},
Countries/**ker** f = {{Germany, England}, {India, China}},
f_1({Germany, England}) = Europe, f_1({India, China}) = Asia.

3.3.2. Tolerance.

A relation *R* in *A* is called a *tolerance* (or a *tolerance relation*) if it is reflexive and symmetric. So tolerance is weaker than equivalence; it does not need to be transitive.

The notion of tolerance relation is an explication of *similarity* or *closeness*. Relations "neighbor of", "friend of" can be considered as examples if we hold that every person is a neighbor and a friend to him(her)self.

As analogs of equivalence classes and partitions, here we have tolerance classes and coverings. A set $B \subseteq A$ is called a tolerance *preclass* if it holds that for all $x, y \in B$, x and y are tolerant, i.e. $<x,y> \in R$. A maximum preclass is called a *tolerance class*. So two tolerance classes can have common elements.

Coverings. Given a non-empty set *A*, a collection (set) Π of non-empty subsets of *A* such that $\cup_{B \in \Pi} B = A$ is called a *covering* of *A*. Given a tolerance relation in *A*, the collection of its *tolerance classes* forms a *covering* of *A*.

Every partition is a covering; not every covering is a partition.

3.3.3. Orderings.

An *order* is a binary relation which is transitive and in addition either (i) reflexive and antisymmetric or else (ii) irreflexive and asymmetric. The former are *weak* orders; the latter are *strict* (or *strong*).

Some terminology: if *R* is an order, either weak or strict, and $<x,y> \in R$, we say that *x precedes y, x is a predecessor of y, y succeeds* (or *follows*) *x*, or *y is a successor of x*. If *x* precedes *y* and $x \neq y$, then we say that *x immediately precedes y* if and only if there is no element *z* distinct from both *x* and *y* such that *x* precedes *z* and *z* precedes *y*. In other words, there is no other element between *x* and *y* in the order.

There is also a useful set of terms for elements which stand at the extremes of an order. Given an order *R* in a set *A*,

1) an element *x* in *A* is *minimal* iff there is no other element in *A* which precedes *x*
2) an element *x* in *A* is *least* iff *x* precedes every other element in *A*
3) an element *x* in *A* is *maximal* iff there is no other element in *A* which follows *x*
4) an element *x* in *A* is *greatest* iff *x* follows every other element in *A*.

Note that greatest (least) element is maximal (minimal) but the opposite is not always the case.

If an order, strict or weak, is also connected, then it is said to be a *total* or *linear* order. The term *partial order* is used for orders in general ('not necessarily total') and is also often used for orders that are not total; this can be confusing but it is a kind of thing that happens fairly often.

Examples. Relations $\geq$ and $=$ on the set **N** of natural numbers are examples of weak order, as are relations $\supseteq$ and $=$ on subsets of any set. The relations $>$ and $\supset$ are examples of strict orders on the corresponding sets. The relations $\geq$ and $>$ are linear orders.

4. Trees.

Here is a first introduction to how one can formalize syntactic tree structures.

We can consider syntactic structures (trees of immediate constituency) as sets of nodes together with some relations (e.g., *is a constituent of*, *is an immediate constituent of*, the order from left to right (*is to the left of*), and one-place properties (unary relations) such as *is a root, is a leaf*). In order to specify what we consider to be the class of possible well-formed trees, we can write down some conditions on these relations and properties.

Which relations to choose as basic?
       One choice we run into if we are trying to decide what relations to consider as most basic is whether to choose "dominates" or "immediately dominates" as more basic, and whether to choose "is to the left of" or "is immediately to the left of" as more basic. Whichever ones we choose, we can define the others.
       On the one hand, it might seem most natural to take the more "local" relations of immediate dominance and immediate precedence as basic. On the other hand, it may turn out to be easier to write conditions on trees using the more general relations.
       So let us begin by considering both a local relation **IDom** "immediately dominates" and the more general relation **Dom** "dominates".
       Dom is the *transitive closure* of **IDom**. What that means is that $<x,y> \in$ **Dom** (or x **Dom** y) iff there is a sequence $x_0, x_1, ..., x_n$ such that $x_0 = x$, $x_n = y$, and for every x_i, $0 \leq i < n$, the pair $<x_i, x_{i+1}> \in$ **IDom.**

Definition: The *transitive closure* of a binary relation R on a set X is the minimal transitive relation R' on X that contains R. Thus $aR'b$ for any elements a and b of X provided that there exist $c_0, c_1, ..., c_n$ with $c_0 = a$, $c_n = b$, and $c_r Rc_{r+1}$ for all $0 \leq r < n$.

Remark: Dom is a partial ordering on the set of nodes. (Convince yourself of that.)

Quick exercise: what is the transitive closure of the relation "is larger than by 1" (i.e. "is the successor of")?

> Thought exercise: Are the following notions well-defined? What has to be checked is whether the presuppositions of the singular "the" are fulfilled, i.e.(i) whether for every relation there is a unique transitive reduction, and (ii) even if the answer to (i) is *no*, one can still ask whether a relation with the properties of **Dom** will always have a unique transitive reduction **IDom**.
>
> And conversely, **IDom** is the *transitive reduction* of **Dom.**
>
> **Definition:** The *transitive reduction* of a binary relation R on a set X is the minimum relation R' on X with the same transitive closure as R. Thus $aR'b$ for any elements a and b of X, provided that aRb and there exists no element c of X such that aRc and cRb.

Question: Does a node dominate itself? As we described it above, **Dom** is not reflexive; in fact it is irreflexive. **IDom** should certainly be irreflexive, and if **Dom** is the transitive closure of **IDom,** then **Dom** must also be irreflexive.

But just as we can define the transitive closure of a relation R, we can also define the *reflexive closure* of a relation R. Informally: Start from R, and add all the "reflexive pairs", i.e. all the pairs <x,x> for x in the domain of R.

Definition: A relation R' is the ***reflexive closure*** of a relation R if and only if

(1) R' is reflexive,

(2) $R \subseteq R'$, and

(3) for any relation R'', if $R \subseteq R''$ and R'' is reflexive, then $R' \subseteq R''$, that is, R' is the smallest relation that satisfies (1) and (2).

We could take the reflexive closure of both **Dom** and **IDom.** But in practice, immediate domination is normally taken to be irreflexive, whereas domination is normally taken to be reflexive. Let us (at least temporarily) use the double arrow $\Rightarrow$ for the transitive reflexive closure of **IDom.** Then the two relations of interest are **IDom** and its transitive reflexive closure $\Rightarrow$. (We are not sure whether terminology in this area has been standardized. Certainly **IDom** is *not* standard.)

A similar relation holds between "is immediately to the left of", which we might write as l, and "is to the left of", $<$. In this case, the relations of interest are unambiguously taken to be intransitive.

++

Alternative approach: Now suppose we take the intransitive "immediate" relations as basic. Then we might start out as follows:

Beginnings of a definition with the "immediate" relations as basic:

Let a tree consist of a finite[1] set **Node** of nodes, with two basic relations defined on **Node**: **IDom** (immediate dominance) and l (immediate left-right precedence). Some conditions we would want to require for **IDom** and l might include the following (you can think of more):

ID0. IDom is a binary relation on **Node: IDom** $\subseteq$ **Node** $\times$ **Node**.

ID1. Every node is immediately dominated by at most one node: For all N $\in$ **Node**, if there are N' and N'' such that **IDom** (N', N) and **IDom** (N'', N), then N' = N''.

ID2. There is exactly one node (the root) that is not dominated by any node: For all N, N' $\in$ **Node,** if there is no N'' such that **IDom** (N'', N) and there is no N'' such that **IDom** (N'', N'), then N = N'.

From these conditions it follows that every node except one is immediately dominated by exactly one node.

But we need other conditions as well. We haven't yet ruled out a loop plus an isolated 'root' node as satisfying the dominance conditions. It would be nice to be able to say that "dominates" is asymmetric: if node A dominates node B, then node B does not dominate

---

[1] Volodja notes that the requirement that the set of nodes be finite is important; infinite trees have different properties and are not usually included in linguists' notion of trees. But the requirement that the set be finite cannot be captured with axioms of first-order logic. (That is an issue we will get to later in the logic section.)

node A. But that would be satisfied for the notion "immediately dominates" even in a loop, as long as the loop has more than two elements.

So it may well be easier to start with the non-immediate notions, $\Rightarrow$ (dominates, not necessarily immediately), and < (precedes). If we require that $\Rightarrow$ be asymmetric, we rule out the possibility of loops. And in fact by using $\Rightarrow$ and <, we can make use of notions already developed for orderings, which makes the constraints much simpler to state.

Beginnings of a definition using the non-immediate relations as basic:

D1. $\Rightarrow$ is a weak partial ordering on **Node.**

O1. < is a strict partial ordering on **Node.** (Note that it is not only 'sisters' that are linearly ordered!)

Interaction condition 1: the Exclusivity condition: For any pair of nodes in **Node**, either they stand in the $\Rightarrow$ relation or they stand in the < relation, and not both.

The non-tangling condition: In any well-formed tree, for any nodes N and N', if N precedes N', then all nodes dominated by N precede all nodes dominated by N'.

Then you can optionally define further relations and properties for trees, such as **Root** as a unary property of a node that is the "top" node of a tree (and you can require that there be exactly one root), **Leaf** as a unary property of a node that is a "bottom" or "terminal" node of a tree (note that linguistic trees are always "upside down", with their root on the top and their leaves on the bottom!). Etc. (If you wanted to keep going, you could try characterizing the relation **C-command**, if you are familiar with it, and any other relations and properties of nodes that you can think of. And you could think about adding labels to the nodes – this can be done in several different ways, either by introducing a new set called **Node-labels** and a binary relation **Label** relating elements of **Node** to elements of **Node-labels**, or just by introducing a family of unary relations **S, NP, VP,** etc. And it is common to distinguish *terminal* and *non-terminal* symbols, and to require that the *leaves* of the tree be labeled by *terminal symbols* and the non-leaves by *non-terminal symbols*.

> If you want to look ahead read about one way of axiomatically characterizing trees, look at Chapter 16, pp 431-448, of Partee, ter Meulen and Wall, from which we have partly borrowed here. Another formalization of trees, in the context of using them to talk about branching time, is in Fred Landman's *Structures for Semantics*, pp 101-120. Another nice formalization, actually a family of them, can be found in Zwicky and Isard (1963), "Some aspects of tree theory", a Mitre Corporation working paper we will copy and distribute. We will return to the axiomatization of trees in a later lecture.

Homework 3.

Chapter 3. pp. 51 – 53, all 5 problems.

6) (optional) Invent a function analogous to the function *f*: Countries → *Continents*. Write down: (a) an equivalence relation which is a kernel of this function, (b) its quotient set, and (c) the corresponding commutative diagram.