RELATIONS

This page is intentionally blank

☺

INTRODUCTION

- *Relations* enable us to investigate connections between members of different sets even if those members are of different types

  - we may have a *set of students* and a corresponding *set of assignment grades*, or
  - a *set of cars* and an associated *set of parking-places*

- More specifically, if we were developing a *Library* system, then, to keep track of *who* has borrowed *what*, we would need to investigate the association between values of type BOOK and values of type PERSON

- Essentially, we need to investigate what are called *ordered-pairs*

ORDERED-PAIRS

- If we have types PERSON and BOOK with:

  Zen and the Art of Motor-Cycle Maintenance : BOOK

  Thomas Tallis : PERSON

  we can create an *ordered-pair* whose first member is the 'book' and whose second member is the 'person'

- The notation for such an **ordered** pairing is:

  Zen and the Art of Motor-Cycle Maintenance $\mapsto$ *Thomas Tallis*

  or, alternatively

  (*Zen and the Art of Motor-Cycle Maintenance, Thomas Tallis*)

- The $\mapsto$ 'arrow' is called a *maplet* and emphasises the *asymmetric* nature of an ordered-pair

- In general, each element of an ordered-pair may be of a different type and hence the type of an ordered-pair cannot, in general, be the type of either component

CARTESIAN PRODUCTS

- Given any two sets, we can, in general, form a third set consisting of *all* the ordered-pairs of elements from those two given sets

- This method of deriving such sets of ordered-pairs is called the *Cartesian product*

- If A is the set {5, 7, 9} and B is the set {3, 5}, then the Cartesian product of the sets A and B is the set of ordered-pairs:

  $$\{(5, 3), (5, 5), (7, 3), (7, 5), (9, 3), (9, 5)\}$$

- The symbol for a Cartesian product is ×, so we can write:

  $$A \times B = \{(5, 3), (5, 5), (7, 3), (7, 5), (9, 3), (9, 5)\}$$

  - $A \times B$ is read as "A cross B"
  - N.B. $A \times B \neq B \times A$

RELATIONS

- Suppose we have:

[COUNTRY]	the set of all countries
[LANGUAGE]	the set of all languages

  then we can define a *speaks* relation between each country and a language spoken in that country, and this relation can be shown as a set of ordered-pairs:

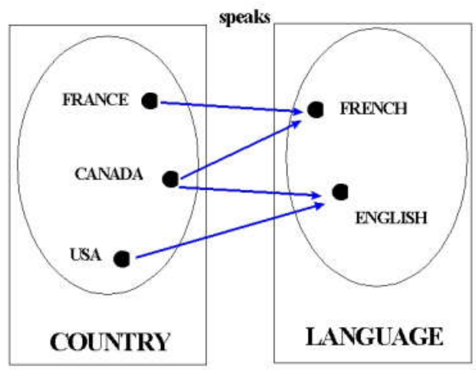
  e.g. {(France, French), (Canada, English), (Canada, French), (USA, English)}

- Note that the elements of the two sets are paired on a *many-to-many* basis; not just a *one-to-one* basis

- Such relations can be illustrated by diagrams called *directed graphs* (or *digraphs*)

SHOWING RELATIONS GRAPHICALLY

- A *digraph* to illustrate the *speaks* relation between the set of all countries and the set of all languages might be as shown alongside



- The set of ordered-pairs corresponding to the relation *speaks* could be declared as

  $$speaks : \mathbb{P}\ (\text{COUNTRY} \times \text{LANGUAGE})$$

- $\mathbb{P}$ (COUNTRY × LANGUAGE)
  - defines all possible sets of ordered-pairs where the first element in each pair is a "country" and the second a "language"
  - represents the **type** of the *speaks* relation

- And we could similarly relate *language* to *country* by declaring a relation

  $$spoken : \mathbb{P}\ (\text{LANGUAGE} \times \text{COUNTRY})$$

  (but see later note on inverse relations)

RELATION SYMBOLS

- The notation just introduced is precise and perfectly acceptable but it is usual to denote a relation using a double-headed arrow: $\leftrightarrow$

- Using this notation the previously introduced relations would be written:

  $$\textit{speaks} : \text{COUNTRY} \leftrightarrow \text{LANGUAGE}$$

  $$\textit{spoken}: \text{LANGUAGE} \leftrightarrow \text{COUNTRY}$$

  - $\leftrightarrow$ is called the *relation symbol*

- For two sets X and Y, writing a type as $X \leftrightarrow Y$ means that, strictly, the type is $\mathbb{P}\,(X \times Y)$

- A statement such as

  $$\textit{speaks} : \text{COUNTRY} \leftrightarrow \text{LANGUAGE}$$

  might be read as:

  "*speaks* relates *country* to *language*"

WAYS OF WRITING RELATIONS

- We have seen that a *relation* is a *set of ordered-pairs*

- We have already seen that ordered-pairs may be represented using either
  - the *maplet* arrow symbol, $\mapsto$, or
  - parenthesised values separated by a comma

- Hence, for the *speaks* relation, we can write the same information in different ways

  e.g. United Kingdom $\mapsto$ English $\in$ *speaks*
  == (United Kingdom, English) $\in$ *speaks*

  and {(France, French), (Canada, English), (Canada, French), (USA, English)}
  == {France $\mapsto$ French, Canada $\mapsto$ English, Canada $\mapsto$ French, USA $\mapsto$ English}

WAYS OF WRITING RELATIONS

- It is, also, acceptable to use English-like language to express a relation:
  - Austria *speaks* German
  - here the name of the relation is used as an *infix* operator between the two values of the ordered-pair
  - in this approach, the relation should be specified as _*speaks*_, with the underscores acting as "placeholders" for the operands

- In general, if R denotes a relation and values x and y are connected through the relation R, then:

$$x \mapsto y \in R == (x, y) \in R$$

but, if we define the relation using: $\_R\_$
then we can write: $x \; R \; y$

ELEMENTS OF RELATIONS

- To discover whether two values are related, it is sufficient to see if the pair is an element of the relation in question

    - If country *Austria* is linked to language *German* in the relation *speak*s where

      $$speaks : \mathbb{P}\ \mathrm{COUNTRY} \times \mathrm{LANGUAGE}$$

      then

      $$Austria \mapsto German \in speaks \text{ will be } \mathbf{true}$$

    - and, if country *Austria* is linked to language *German* in the relation _*speak*s_ where

      $$\_speaks\_ : \mathbb{P}\ \mathrm{COUNTRY} \times \mathrm{LANGUAGE}$$

      then

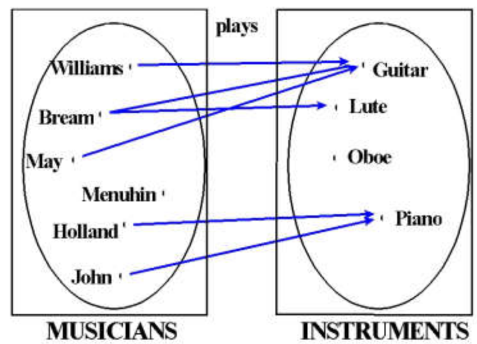
      $$Austria\ speaks\ German \text{ will be } \mathbf{true}$$

TO-SET AND FROM-SET

- For a relation X$\leftrightarrow$Y between the elements of the sets X and Y, then we say

  X is the *from-set* (or *source)*, and
  Y is the *to-set* (or *target*)

- Consider the relation *plays*, as illustrated alongside



- In the *plays* relation as shown,
  - the from-set has some element(s) not related to any element of the to-set ; likewise
  - the *oboe* of the to-set is not apparently played by any element of the from-set

DOMAIN AND RANGE

- When dealing with relations, we are interested, usually, only in those elements in the from-set and to-set which are actually related

- The subset of elements in the from-set which are related to *at least one* element of the to-set is called the **domain** of the relation

  e.g. the **domain** of the *plays* relation is:
  {Bream, Holland, John, May, Williams}

- In like manner, the subset of elements in the to-set, which are related to some element in the from-set, is called the **range** of the relation

  e.g. the **range** of the *plays* relation is: {Guitar, Lute, Piano}

- Usually we abbreviate *domain* and *range* to **dom** and **ran**, so the above examples can be written:

  dom *plays* = {Bream, Holland, John, May, Williams}
  ran *plays* = {Guitar, Lute, Piano}

RELATIONAL INVERSE

- It is often useful to look at a relation ‘the other way round’

- The “infix” *_plays_* relation shows which instruments are played by particular musicians

- To show which musicians play particular instruments, we could define an ‘other way round’ infix relation called *_is-played-by_*

  - as part of the *_plays_* relation we have:

    Holland plays piano

    and so would expect:

    piano is-played-by Holland

- By reversing ***all*** the ordered-pairs in a relation we obtain another relation known as the ***inverse*** of the original relation

- Thus *_is-played-by_* is the inverse relation to *_plays_* and this is written symbolically as:

$$\_is\text{-}played\text{-}by\_ = \_plays\_^{-1} \qquad (\text{ or, } \_plays\_^{\sim})$$

RELATIONAL IMAGE

- If we are given any subset of the *domain* elements of a relation, then the subset of corresponding values from the *range* of the relation is given by the *relational image*

- From our *plays* relation, the set of instruments played by *Williams* or *Bream* is {*guitar*, *lute*} and this can be written symbolically, using the *relational image*, as

$$plays \llparenthesis \{Williams, Bream\} \rrparenthesis = \{guitar, lute\}$$

- As a further example, for the *speaks* relation, an expression which specifies the languages spoken in France and Switzerland is:

$$speaks \llparenthesis \{France, Switzerland\} \rrparenthesis$$

RELATIONS ARE SETS

- Since a relation is a set of ordered-pairs it can be manipulated with the usual set operations:

- For the relation *plays* where:

  $$plays : \text{MUSICIANS} \leftrightarrow \text{INSTRUMENTS}$$

  if we add the extra information that *Oistrakh* plays the *violin*, we can write:

  $$plays' = plays \cup \{(Oistrakh, violin)\}$$

  $$\text{or,} \quad plays' = plays \cup \{Oistrakh \mapsto violin\}$$

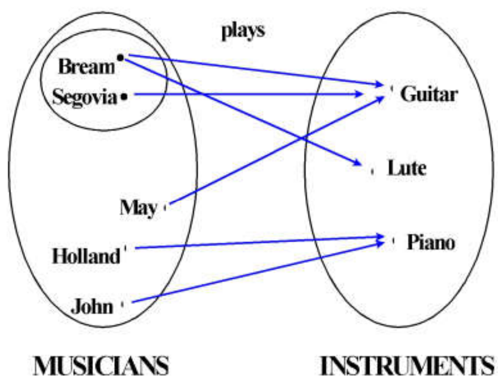
  Here the prime annotation $'$ indicates an "updated" value

DOMAIN RESTRICTION

- It is often convenient to consider a relation restricted to only *part* of the original domain

- Consider the *plays* relation shown below where:

  $$Classical\_musicians == \{Bream, Segovia\}$$



- If we are only interested in that part of the domain which includes the set of "classical" musicians, we could define a smaller (restricted) relation called, say, *Classical_plays* by:

  $$Classical\_plays == Classical\_musicians \lhd plays$$

DOMAIN RESTRICTION

- Note that:

  $Classical\_musicians \lhd plays \quad \Leftrightarrow$

  $(Classical\_musicians \times INSTRUMENTS) \cap plays$

- $\lhd$ is the *domain restriction* operator and can be defined by:

$$S \lhd R == \{x{:}X;\ y{:}Y \mid x \in S \wedge x \mapsto y \in R\}$$

where **dom** R : X and **ran** R : Y

- *Domain restriction* restricts the ordered-pairs of a relation so that only those pairs where the first element is contained in a particular, restricted, set of the original domain are included

DOMAIN CO-RESTRICTION

- Suppose further that the original domain of our relation, *plays*, comprises two **disjoint** sets (these could be *Non_Classical_musicians* and *Classical_musicians*) then if, as above, *domain restriction* has been used to isolate one of those two disjoint sets, the other disjoint set is given by *domain co-restriction* (aka *domain subtraction* or *domain anti-restriction*)

- If we assume that the musicians of our *plays* relation are either "classical" or "pop" musicians but **cannot be both** then domain co-restriction allows us to restrict the *plays* relation to *non-classical* musicians of the original domain thus:

$$Classical\_musicians \mathbin{⩤} plays$$

- Which is equivalent to something like:

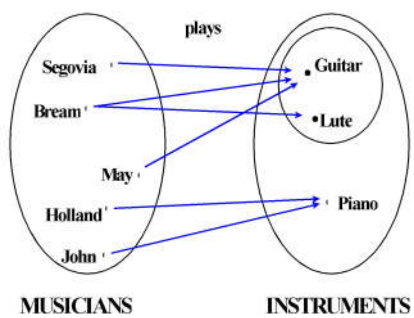
$$plays \setminus (Classical\_musicians \times INSTRUMENTS)$$

RANGE RESTRICTION

- To complement the *co-restriction* and *restriction* operations for *domains* there are similar operations that apply to *ranges*

- Suppose we wish to concentrate our attention on the set of 'plucked' instruments where

$$Plucked\_instruments == \{Guitar, Lute\}$$



- To confine the *plays* relation to those ordered-pairs whose second members are in the *Plucked_instruments* set, we write:

$$plays \rhd Plucked\_instruments$$

which is equivalent to:

$$(\text{MUSICIANS} \times Plucked\_instruments) \cap plays$$

RANGE CO-RESTRICTION

- *Range co-restriction* (aka *range subtraction* or *range anti-restriction*) draws attention to those pairs in a relation where the second elements are NOT members of some set of interest in the range of the relation

- Thus in our *plays* relation where we have separated the members of *Instruments* into *Plucked_instruments* and others:

  - the expression:

    $$plays \rhd Plucked\_instruments$$

    restricts the relation to those pairs where the second element is neither *Guitar* nor *Lute*

- $plays \rhd Plucked\_instruments \Leftrightarrow$

  $$plays \setminus (MUSICIANS \times Plucked\_instruments)$$

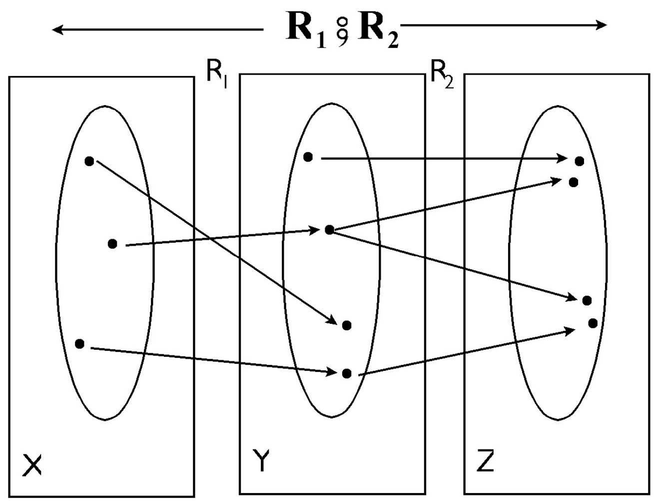
RELATION COMPOSITION

- If we have two relations R_1 and R_2 where the to-set of R_1 is of the same type as the from-set of R_2 then we can form a third relation R_3, say, which is the *composition* of R_1 and R_2:

$$R_1 : X \leftrightarrow Y \text{ and } R_2 : Y \leftrightarrow Z$$

- The relation R_3, formed by the composite relation "R_1 then R_2", is called the *forward composition* of R_1 with R_2 and is denoted by:

$$R_1 \fatsemi R_2 : X \leftrightarrow Z$$



- The symbol $\fatsemi$ is often called the *fat semi-colon*

REPEATED COMPOSITION

- If a relation relates domain values of some type to range values of the **same** type, the relation is said to be *homogeneous*

- The composition of homogeneous relations is often called *repeated composition*

- Clearly any homogeneous relation can be composed with itself (*self-composition*):

  if $R : X \leftrightarrow X$ then $R \fatsemi R : X \leftrightarrow X$

- If we imagine _*borders*_ to be a relation between countries which share a border then:

  $$\_borders\_ : \text{COUNTRY} \leftrightarrow \text{COUNTRY}$$

  (e.g. France *borders* Germany)

- Writing the composition *borders* $\fatsemi$ *borders* implies two countries each share a border with a third:

  France *borders* $\fatsemi$ *borders* Austria

REPEATED COMPOSITION

- Such repeated composition is often abbreviated with a *power* symbol:

    - France *borders* ^{3} Hungary

      is equivalent to:

      France *borders* ⨾ *borders* ⨾ *borders* Hungary

      (which is, itself, a shortened form of:

      France *borders*
      Germany *borders*
      Austria *borders* Hungary)

- In general, $x \;\; R^{+} y$, implies there is a repeated composition of relation _R_ which relates x to y

  For example: "France *borders*$^{+}$ India" implies France and India are on the same landmass

IDENTITY RELATION

- The *identity* relation is symbolized by **id** and maps all elements of a set onto themselves:

$$\mathbf{id}\ X == \{\ x : X \bullet x \mapsto x\}$$

e.g.	if	$X=\{p, q, r\}$
	then	$\mathbf{id}\ X = \{(p, p), (q, q), (r, r)\}$
	or	$\mathbf{id}\ X = \{p \mapsto p,\ q \mapsto q,\ r \mapsto r\}$

- It is, perhaps, worth noting that it is possible to define some of the operations on relations in terms of the identity relation:

e.g.	$S \lhd R == (\mathbf{id}\ S) \mathbin{⨾} R$
and,	$R \rhd S == R \mathbin{⨾} (\mathbf{id}\ S)$

SUMMARY of SYMBOLS

- P, Q, S are sets; p : P; q : Q and R : P $\leftrightarrow$ Q

$P \times Q$	*the set of ordered-pairs of elements from P and Q respectively*
$P \longleftrightarrow Q$	*the set of relations from P to Q;* (*same as* $\mathbb{P}(P \times Q)$)
$p \; R \; q$	*p is related by R to q;* (*equivalent to* $(p, q) \in \_R\_$)
$p \mapsto q$	*ordered pairing of p and q*; (*equivalent to* (p, q))
$\{p_1 \mapsto q_1, p_2 \mapsto q_2, \ldots\ldots\ldots p_n \mapsto q_n\}$	*the relation* $\{(p_1, q_1), (p_2, q_2), \ldots, (p_n, q_n)\}$ *relating* p_1 *to* q_1, p_2 *to* q_2, ..., p_n *to* q_n
dom R	*the domain of a relation*
ran R	*the range of a relation*
R^{-1}	*the inverse of a relation R;* (*also sometimes written* $R^{\sim}$)
$R \llparenthesis S \rrparenthesis$	*the relational image of S in R*

- Below Q and S are sets; R, R_1, R_2 are relations

$S \lhd R$	*the relation R, domain restricted to the set S*
$R \rhd S$	*the relation R, range restricted to the set S*
$S \ntriangleleft R$	*the relation R, domain co-restricted to the set S*
$R \ntriangleright S$	*the relation R, range co-restricted to the set S*
$R_1 \fatsemi R_2$	*the (forward) composition of* R_1 *with* R_2
R^+	*the repeated self-composition of R with itself*
id Q	*the identity relation on Q*