APPENDIX: GENERAL PREREQUISITES

In the body of this text we have assumed familiarity with several topics not specifically connected with classical logic. In this appendix we present a brief introduction to these topics.

1. Sets and sequences

Intuitively speaking, a set is a set *of objects*. Thus a set is a single object corresponding to what are possibly many different objects. If S is the set of objects s_1, s_2, ... and T the set of objects t_1, t_2, ..., then the two sets are taken to be the same if the objects are the same, i.e. if each s_i is a t_j and each t_j an s_i. We say that the object s is *a member of* the set S - in symbols, $s \in S$ - if S is a set of objects s_1, s_2, ..., of which one is s. We use '{s_1, s_2, ...}' for the set of objects s_1, s_2, ... and {s: ---} for the set of objects s satisfying the condition --- (some other obvious variants of this notation will also be used). We assume that there is a *null set* {} - also written as '$\emptyset$' - with no members .

If every members of the set S is a member of the set T, then we say that S is a subset of T(or sometimes that S is included in T or that T contains S) and we write '$S \subseteq T$'. If S is included in T, but not identical to it, we say that S is a proper subset of T (or that S is properly contained in T, or that T properly includes S) and write $S \subset T$. Sometimes it is useful to know the original set from which a subset is selected. A sub-set (with a hyphen) with domain D and subset S (written (S/D)) comprises two sets, with the first containing the second. For example if A,B,C are the sets of all animals, birds, and crows, respectively, then the sub-set (A/C) can thought of as the set of animals that are crows; and (B/C) as the set of birds that are crows. If the domain D is understood we shall sometimes identify a sub-set S/D with its subset S. Terminology for subsets is extended in an obvious manner to sub-sets. We may say, for example, that sub-set S/D is included in sub-set S'/D' if $S \subseteq S'$ and $D \subseteq D'$. Two sub-sets S/D and S'/D' are said to coincide if S=S'.

The power set of S, written $\mathcal{P}(S)$, is the set of all subsets of S. The intersection of sets S and T, written $S \cap T$, is the set of all objects that are members of both S and T. The union of S and T, written $S \cup T$, is the set of all objects that are members of S or T (or both). The difference of S and T, written S-T, is the set of all objects that are members of S but not T. Union and intersection are usefully generalized. If the members of S are sets, then $\bigcup S$ is the set of all members of members of S, i.e., $\bigcup S = \{x : x \in s \text{ for some } s \in S\}$, while $\bigcap S$, when S is nonempty, is the set of all objects that are members of every member of S, i.e., $\bigcap S = \{x : x \in s \text{ for all } s \in S\}$.

We turn to sequences. Sequences, like sets, make one object from many. But when s is a sequence of objects $s_1,...,s_n$ and t a sequence of objects $t_1,...,t_n$, then the two sequences are taken to be the same when the corresponding terms are the same, i.e. when the sequences are of the same length and $s_1 = t_1$, $s_2 = t_2$, ... It is assumed that any sequence is either finite (i.e. a sequence of objects $s_1,...,s_n$ for $n > 0$) or denumerable (i.e. a sequence of objects s_1, s_2, ...,s_k, ..., where there is a term s_k for each natural number k = 1, 2, ...). A finite sequence is also called a tuple and a sequence of n objects an n-tuple (2-tples and 3-tples are also known as (*ordered*) *pairs* and *triples*). We use '(s_1, s_2, ...)' for the sequence of objects s_1, s_2, ... If s is a finite sequence ($s_1,...,s_n$) and t a finite or denumarable sequence (t_1,t_2,...), then we take the concatenation of s and t - written as s^t - to be the sequence (s_1,..., s_n, t_1, t_2,...) consisting of the terms of s followed by the terms of t. If S and T are any sets, then the *Cartesian product* SXT is the set of all pairs (s,t) such that $s \in S$ and $t \in T$. More generally, if $S_1,...,S_n$ are any sets, then S_1X...XS_n is the set of all n-tuples $(s_1,...,s_n)$ such that $s_1 \in S_1,...,s_n \in S_n$. In particular, S^n is the product SXSX ... X S(n times) of all n-tuples of members of S.

2. Relations and functions

A relation is something that can hold or fail to hold among certain objects. The relation of being the father of, for example, holds between Johann Sebastian Bach and Johann Christian Bach, but not between John Cage and John Lennon, while the relation of being between holds of New York, Washington and Boston though not of Boston, New York and Washington. We make three assumptions about relations. First, each relation is taken to hold of a fixed number $x_1,..., x_n$ of objects (which is not to say that no two of the objects can be the same). The number n of objects of which a relation holds is called its *degree* or *adicity*. Thus the father-of relation is of degree 2, while the betweeness relation is of degree 3. Relations of respective degrees 1, 2, 3 and n are said to be *unary*, *binary*, *ternary*, and *n-place*. Second, relations of given degree that hold of the same objects are taken to be the same. Third, given any set X of n-tples, we assume that there is an n-place relation that holds exactly of the objects $x_1,..., x_n$, for which $(x_1,..., x_n) \in X$. In the light of the last two assumptions, we may identify any n-place relation, for given n, with the set of n-tples of objects of which it holds.

If R is an (n+1)-place relation, we write ‘ $Rx_1...x_nx$’ or ‘$x_1...x_nRx$’ to indicate that R holds of $x_1, ... , x_n$, x. Of course, if R is identified with a corresponding set of n-tples, then we may also write ‘ $(x_1,...,x_n,x) \in R$’.

3.Properties of binary relations

(Omitted).

4. Induction

Induction is a mathematical method of proof that is used to establish that all objects in a given class have a certain property. The most familiar kind of induction is number induction. This makes use of the following principle.

P1. All natural numbers have the property P if:
i. the number 0 has property P; and
ii. k+1 has P whenever k has P.

In applying P1, we call the step used to establish i the basis and the step used to establish ii the inductive step. The assumption that is made in the inductive step is called the induction hypothesis. For example, to prove that the sum of the first n positive integers is n(n+1)/2 it is sufficient to show:

basis: 0=0(0+1)/2, and

inductive step: if the sum of the first k numbers is k(k+1)/2, then the sum of the first k+1 numbers is (k+1)(k+1+1)/2.

Sometimes it is useful to be able to call on a slightly stronger induction hypothesis. This is permitted by the principle of course of values induction for numbers.

P2. All natural numbers have the property P if:
i) the number 0 has property P;
ii) k+1 has P whenever all numbers less than or equal to k have P.

It should be clear that the principles P1 and P2 are correct. The basis establishes that the number zero has the property. If number 0 has the property, then the inductive step establishes that the number 1 does. If numbers 0 and 1 have the property, then the inductive step establishes that the number 2 does. And so on. For any particular number k, one application of the basis and k applications of the inductive step show that k has the property. So all numbers have the property.

Let us give a simple example of proof by number induction (which will also serve to illustrate how such proofs should be laid out). We shall establish the identity:

(*) $0 + 1 + ... + n = n(n + 1)/2$ for any natural number n.

Proof By number induction.

Basis (i.e. the case n = 0). We need to show $0 = 0.(0 + 1)/2$. But RHS. = 0.0/2 = 0 = LHS.

Inductive Step (i.e. the case n = k+1). Suppose that the equation holds for n = k (this is the inductive hypothesis). We need to show that the equation holds for n = k+1. But:

$$0 + 1 + ... + k + (k+1) = k(k+1)/2 + (k+1) \text{ (since } 0 + 1 + ... + k = k(k+1)/2 \text{ by IH)}$$
$$= [k(k+1) + 2(k+1)]/2$$
$$= (k+1)(k+2)/2, \text{ as required.}$$

The following is a more streamlined version of the above proof (in which the supposition of IH and the 'flags' are left implicit):

n = 0 RHS = 0.0/2 0 = LHS.

n = k+1 $0 + 1 + ... + k + (k+1) = k(k+1)/2 + (k+1)$ (by IH)

$$= (k+1)(k+2)/2.$$

Exercises on number induction

1. Prove by number induction that $1 + 3 + 5 + ... + (2n + 1) = (n+1)^2$.
2. Prove by number induction that $0 + 2 + 4 + ... + 2n = n(n+1)$.
3. Derive the conclusion that $0 + 1 + 2 + ... + n = n(n+1)/2$ from the results of exercises 1 and 2 without using induction.
4. Let X be a set with n elements. Show by number induction that:
   (i) there are 2^n subsets of X; and
   (ii) there are n! ways of arranging the elements of X into a sequence of n elements.
5. Prove by course-of -values number induction that every natural number n > 1 is a product $p_1.p_2. ... p_n$ of prime numbers $p_1, p_2, ... p_n$, $n \geq 1$.
6. Show that course-of-values induction for numbers may be stated in the form:
   every number has P if a number has P whenever every preceding number has P.

(The basis gets swallowed up in the inductive step).

It should be clear that the kind of principle typified by P1 and P2 above does not merely apply to the natural numbers. The reason that they hold is that any positive integer can be reached from the number one by adding one finitely many times. Let us see how this feature of the natural numbers may be generalized. Given a set S and relations binary relations $R_1,R_2,...$ that hold between subsets of S and members of S, let us say that S is generated from the set B by means of the relations $R_1,R_2,...$ if the following three conditions are satisfied:

(i) B is a subset of S;

(ii) S is closed under the R_i's, i.e. if T is a subset of S and T R_i s then $s \in S$.

(iii) No other objects belong to S; i.e., any member of S can be obtained by repeated applications of (i) and (ii).

If the conditions (i)-(iii) are satisfied, then S is called a <u>generated set</u>, the members of B the <u>basic elements</u>, and $R_1, R_2, \ldots$ the <u>generating relations</u>. For example, the set of natural numbers is generated from {0} by the relation that holds between {0, 1, ..., n} and n+1. They are also generated from the prime numbers and 0 by the product relation (which holds between {n, m} and nm). According to <u>Genesis</u>, the set of human beings is generated from the set {Adam, Eve} by the relation **are-parents -of** (which holds between a pair of parents and their biological children).

Notice that there may be one basic element, as in the first example, several basic elements, as in the third example, or infinitely many basic elements ,as in the second example. The generating relations may be "single-valued" (always relating a set to a single object) as in the first two examples or "multi-valued" (sometimes relating a set to several objects) as in the third example. Notice also that it might be possible to generate a given set (such as the natural numbers) in several different ways.

According to the definition above, the generating relations are binary relations between subsets of S and elements of S. When the subsets are finite, it is often more natural to think of the generating relations as holding between certain members of S (those constituting the subset) and another member of S. Clause (ii) above then takes the form:

(ii') if $s_1, s_2, \ldots, s_n$, are members of S and $s_1, s_2, \ldots, s_n R_i s$, then s is a member of S.

For example, the set of human beings (assuming the truth of <u>Genesis</u>) is generated from {Adam, Eve} by the relation that holds between x, y and z when x and y are the biological mother and father of z.

<u>Exercises</u>

1. What is a natural basis and generating relation (or set of generating relations) for the set of *positive* integers 1, 2, ...? For the set of all integers - negative, zero and positive? For the set of arabic numerals?

2. Suppose that S is the set of nonnegative rational numbers, B the set of natural numbers, and R the relation that holds between {x, y} and z when z = x/y.

(i) Show that each number in S above can be generated in infinitely many different ways.

(ii) Replace R with the relation that holds between {x, y} and z when x and y have no common divisor and z = x/y. Now show that each number in S but not B (i.e. each properly rational number) can be generated in exactly one way.

3. (i) Show that a set S is generable from itself by means of any relations.

(ii) Let R be a relation which holds between any subset and any element of S. Show that X can be generated from any basis whatever by means of R.

(iii) Hence show that any set containing at least one element can be generated in at least two ways.

4. Show how a set of generating relations can be replaced by a single generating relation. (Hint: take the single generating relation to be the union of the given generating relations.)

Suppose that the set S is generated from basis B by means of the generating relations

R_1,R_2,.... . There is then a principle of induction corresponding to this method of generating the elements of S:

P3. All members of S have the property P if:

i. all basic elements have P; and

ii. the generated relations preserve possession of the property, i.e. x has the property whenever all members of T have the property and R_i x for some generating relation R_i.

To see why this principle holds, note that all basic elements have the property by i, that all elements generated from basic elements have the property by ii, that all elements generated from those elements and the basic elements have the property by ii again; and so on. So by iii in the definition of generation, it follows that all elements of the generated set have the property.

We may extend the terminology for number induction to the more general case. Thus an argument that proceeds on the basis of P3, and establishes that all elements of the generated set have a given property by establishing i and ii above, is a called a *proof by induction*. The whole argument is called an *induction*; its first premiss i is the *basis*; and its second premiss ii the *inductive step.*

Frequent appeal will be made to two special cases of P3 throughout the book - one concerning formulas and the other concerning theorems. Let us deal with each in turn. We may take the formulas of sentential logic (as described in §1.1) to be generated from the set {p_0, p_1, ...} of sentence-letters by means of the syntactic relations corresponding to negation and disjunction (in the case of negation, the relation is the one that holds between {A} and B when B is the negation -A of A and, in the case of disjunction, the relation is the one that holds between {A, B} and C when C is the disjunction (B v C) of A and B.)

Thus the principle of induction in this special case takes the form:

P3 (for formulas). All formulas have the property P if:

i. all sentence-letters have P; and

ii. the negation of any formula with the property has the property and the disjunction of any formulas with the property have the property.

To see this principle at work, let us consider the proof of the following result:

Theorem Each formula of sentence logic contains an even number of occurrences of brackets.

Proof. Let n(A) be the number of occurrences of brackets in A. Then we must show that n(A) is even for all formulas A. The proof is by formula induction.

Basis (the case in which A is a sentence-letter p). In this case, n(A) = 0, which is even.

Inductive Step (the case in which A is either of the form -B or (B v C)).

A = -B Suppose that n(B) is even (the inductive hypothesis). To show that n(-B) is even. But n(-B) = n(B), which is even by IH.

A = (B v C). Suppose that n(B) and n(C) are even (the IH). To show that n((B v C)) is even. Now n((B v C)) = 2 + n(B) + n(C). But n(B) and n(C) are even by IH; and so n((B v C)) is also even.

Exercises (on formula induction)

1. Let l(A) be the number of occurrences of left-hand brackets in A, r(A) the number of occurrences of right-hand brackets in A, and d(A) the number of occurrences of ‘v’ in A.

(i) Show by formula induction that l(A) = r(A).

(ii) Show by formula induction that d(A) = l(A).

2. (i) Show by formula induction that the first bracket of any formula with brackets is a left-hand bracket and that the last bracket of any such formula is a right-hand bracket.
(ii) Let A^r be the result of writing A backwards (so when A is the sequence of symbols $s_1\ s_2 \ldots\ s_n$, A^r is the sequence $s_n\ s_{n-1} \ldots\ s_1$). Deduce from (i) that if A is a formula with brackets then A^r is not a formula at all.
3. Use a simple induction on formulas to establish:
(i) The first symbol of any formula must either be a '-' or a LH bracket or a sentence-letter;
(ii) The last symbol of any formula must either be a RH bracket or a sentence-letter.
Then use another induction to establish the following additional facts about the possible patterns of occurrence of symbols that may occur in formulas:
(iii) A LH bracket must be followed by another LH bracket or by a sentence letter of by a '-';
(iv) A RH bracket cannot be followed by a LH bracket or a sentence-letter of a '-';
(v) A 'v' must be followed by a LH bracket, a sentence-letter of a '-';
(vi) A '-' must be followed by a '-' or a LH bracket or a sentence-letter;
(vii) A sentence-letter cannot be followed by another sentence-letter of a LH bracket or a '-'.
4. (Concerning the results under exercise 8, but not induction) .
(i) Show that the results under exercise 8 cannot be strengthened.
(ii) Show that an expression may conform to all of the conditions (i) - (vii) listed under exercise 9 and still not be a formula. (Hence a machine which scanned expressions from left to right and which could only retain the memory of the previously scanned symbol would not be able to test for formulahood.).
(iii) Say that an expression is *admissable* if it is a subexpression of some formula. For n = 1, 2, ... , say that the expression E is *n-safe* if any subexpression of E of length n is admissable. Show, for each n, that there is some n-safe expression which is not a formula. (Hint: consider a sequence of n RH brackets.)

(Material on theorem induction to be included later).

Generation Sequences

Suppose that we wish to establish that a given object is indeed a member of a generated set (that a given expression, for example, is a formula). Then we may do this by means of a *generation sequence*. This is a sequence of objects $x_1, x_2, \ldots, x_n$, $n \geq 1$, which is such that , for each k = 1, 2, ..., n, x_k is either an element of X_0 or is generated from previous members of the sequence by means of one of the generating relations. It should be clear that the generated elements are exactly those that appear as the last term of some generation sequence.

As examples of generation sequences, we have the sequence Adam, Even, Cain of human beings, the sequence 0, 1, ..., n of natural numbers, and the sequence p_1, p_2, (p_1 v p_2), -(p_1 v p_2) of formulas. If need be, we may annotate each entry within a generation sequence in order to indicate what justifies its presence within the sequence. Thus annotating the sequence of formulas above in an obvious manner, we obtain:

(1) p_1 (a sentence-letter);
(2) p_2 (a sentence-letter);

(3) $(p_1 \vee p_2)$ (from (1) and (2) by disjunction);
(4) $-(p_1 \vee p_2)$ (from (3) by negation).

(Material on other forms of induction to be included later).