Klaus Bung (1971):

'The concept of partial order in programmed language instruction and the freedom of the consumer'.

In: PROGRAMMED LEARNING AND EDUCATIONAL TECHNOLOGY,

Vol 8, p 22-33 and p 122-124. Sweet and Maxwell, London, England

Part 1 (Vol 8, p 22-33)

THE CONCEPT OF PARTIAL ORDER IN LANGUAGE PROGRAMMING AND THE FREEDOM OF THE CONSUMER

KLAUS BUNG

Dept. of Linguistics, Cambridge University

Abstract: This is the first part of an article. Language teaching consists of developing in the student the ability to manipulate efficiently the mechanics of the target language and of teaching him to use the language mechanics in life-like situations. The former aspect of language teaching can be dealt with by means of programmed instruction. All language skills (in the above sense of "manipulation") can be described by rules. The term rule is extended to cover also so-called exceptions and directions to perform purely imitative tasks. All rules can then be interpreted as mappings of the following types:

(1) $\varphi : A \rightarrow B$ (mapping of one set into another: paired associate learning, copying)

(2) $\psi : A \rightarrow p(A)$ (mapping of one set into its partition: classificatory skills).

Rules can be formulated as quasi-algorithms (QAs). A QA must be explicit, *i.e.* it must contain only elementary operations. The concept of "elementary operation" in QAs is relative and depends on the prior knowledge/skill of the addressee. Between pairs of QAs bilateral relations are established, expressed as $a \rightarrow b$ (*i.e.* a must precede b), in accordance with the principle: all operations contained in a QA must either be explicitly described by a preceding QA or be contained in the addressee's initial repertory of skills. This condition is satisfied by imposing a partial order on the set of all QAs. The pedagogical grammar is defined as a partially ordered set (POS) of QAs satisfying the above condition. A hierarchy of POSs is developed, where a teaching step T is an ordered set of elements, an exercise E is either a strictly ordered set of Ts or a "totally unordered" set of Ts. An assembly of exercises, Q, is the partially ordered set of exercises E realising a given QA. A section S is the POS of Qs dealing with similar subject-matter. A component C is the POS of sections S all of which demand responses in the same medium (graphic or acoustic). A diagram D is the "totally unordered set" of components representing "communication skills." A task analysis at the level of sections is given. It is then demonstrated that a teaching program which is conceived as a POS of elements maximises the freedom of choice for the learner and the teacher, whereas this freedom is minimised both in conventional teaching programs (linear and branching alike) and in traditional classroom teaching.

BACKGROUND

PRACTICAL work in the field of programmed instruction in languages is almost as old as the praxis of programmed instruction in general. The outstanding old practitioners in the field, Morton (1959), Marty (1962), Carroll (1963) and Valdman (1966) have expressed their more recent views in Mueller (ed.) (1968). Work in the field has inherited a strong anti-theoretical bias from Skinner (*e.g.* 1950). Whatever theory there was (Skinner, 1957) as a basis for language programs has been shown to be largely inadequate for explaining the phenomena of language learning (Chomsky, 1959). As a result, the feasibility of programmed language instruction has been seriously questioned (Spolsky, 1968). From this background and that of practical teaching experience, the author developed the AALP-Theory (Theory of Adaptive Algorithmic Language Programming), which aims at explicitly describing, and relating in one coherent system, all teaching and learning functions, such as program construction (*e.g.* task analysis, design of the pedagogical grammar, construction of exercises), program evaluation and program use (*e.g.* adaptivity, teacher function). The present paper does not directly take issue with any of the earlier writers on programmed language instruction. It operates strictly within the framework of the AALP-Theory and tries to sharpen and systematically explore some of the structural concepts used implicitly in earlier publications on this theory, namely the concepts of partial order and of hierarchy. Recently an attempt has been made (1970b) * to incorporate the AALP-Theory into the framework of the six didactic variables of Heimann (1962) and Frank (1969, Vol. 1 pp. 42–59). In this paper certain elementary concepts from other disciplines have been used, and for the newcomer to these fields the following introductory texts will be helpful: theory of sets (Stoll, 1961, and Goodstein, 1963), of algorithms (Glushkov, 1966), of quasi-algorithms (Landa, 1968 and 1969; Lánský, 1969; Bung, 1969c), of graphs (Ore, 1963). Since writing this paper, Banerji (1969) has come to the author's attention.

* References to the author's own publications are given by year only.

The Language Program as a Hierarchy of Partially Ordered Sets

1. *Rules as mappings*

Foreign language teaching implies two tasks. The teacher has to teach the student how to manipulate the mechanics of the foreign language efficiently and how to apply his manipulatory ability in life-like situations. By " manipulation of language mechanics " we mean the ability of the student to produce and understand sentences and strings of sentences which are grammatically correct. This ability can be taught by means of programmed instruction and the following discussion is confined to this ability. The task of teaching the use of language in life-like situations is reserved for the teacher (1969b).

Let us assume, the pupil is to learn how to produce the sentence "Ich weiß, daß ich nichts weiß" and other sentences of the same type. In order to be able to perform this task, he must have certain items of prior knowledge (prior skill). For instance, he must know that, in a German subordinate clause the verb always occupies the last place, that daß-clauses are subordinate clauses, how to conjugate the verb "ich weiß," etc. All items of prior knowledge can also be interpreted as prior skills. All skills can be described by rules. We use the term *rule* in such a way that it also comprises the " special prescriptions " normally called " exceptions " in traditional grammar (*cf.* 1969c). All rules in language teaching can be interpreted as mappings. It seems that two types of mappings are sufficient to describe exhaustively all types of rule (and hence all types of skill).

Assume a pupil is to learn an unambiguous unidirectional association. For instance, he is to learn the translation of a word in his native language for which the target language has only one "equivalent." Or he is to learn the plural which belongs to the singular form of a German noun, or the "arbitrary" pronunciation which is associated with the orthographic version of an English word. In all these cases we are faced with an (unambiguous) mapping. An input word provided by the teacher, the teaching program, the dictionary, the thoughts of the pupil is to be mapped onto a specific output word. During this process the input word as a whole has to be identified but there is no checking of distinctive features of the input word and no corresponding association of the input word with a certain class. Thus this mapping procedure corresponds to an algorithm which does not contain any discriminators (predicates). A special case of this type of mapping is the one in which a pupil is to copy a given utterance, in speaking or in writing. Since complete identity of the copied object with the given object is never possible but there are always differences, however minute, between model and copy, the case of copying must also be interpreted as a mapping of one object onto another. If we denote the unambiguous mapping function with φ, the set of input words with A and the set of associated output words with B, then we can represent this type of rule as follows:

$$\varphi\colon\ A \rightarrow B$$

A second type of rule describes the skill by means of which a pupil partitions a set of input words into several sub-sets (classes). This is, for instance, the case when the pupil is to decide whether a given German word which contains the sound /s/ is to be spelt with "s," with "ss" or with "ß." In this case the input set, consisting of all German words which contain the sound /s/, is to be divided up into the three sub-sets whose elements are the words with "s," "ss" and "ß" respectively. In other words, each element of the input set is assigned unambiguously to one of three classes. The class membership then determines how the sound /s/ is to be spelt in the given word. We are dealing with a similar case when the pupil learns where to place the finite verb in German. In that case, the input set is the set of possible sentence types in German (*e.g.* declarative sentence, subordinate clause, yes-no question, w-question, imperative clause, conditional clause without "wenn," etc.) and the output set consists of the sub-sets consisting of those sentences which are characterised by any one of the possible verb positions (first, second and last place). For this second type of rule, input set and output set are related in as far as the output set consists of classes (sub-sets) of the partition of the input set. We denote the partition of A with p(A). Thus the input set is mapped onto its partition. If we denote the partitioning function with ψ, then we can represent this type of rule as follows:

$$\psi\colon\ A \rightarrow p(A)$$

In order to establish classes (partitions), certain criteria (features) of the input words have to be checked, *i.e.* the algorithm which describes the partitioning function ψ must contain at least one discriminator.

2. *Quasi-algorithms*

The presence of certain criteria in an input word can be checked by means of an explicit procedure, which we will call quasi-algorithm. (The distinction between algorithm and quasi-algorithm is explained in 1969c.) (1) (p. 26) shows an example of such a procedure, which determines the spelling of the sound /s/ in German words.

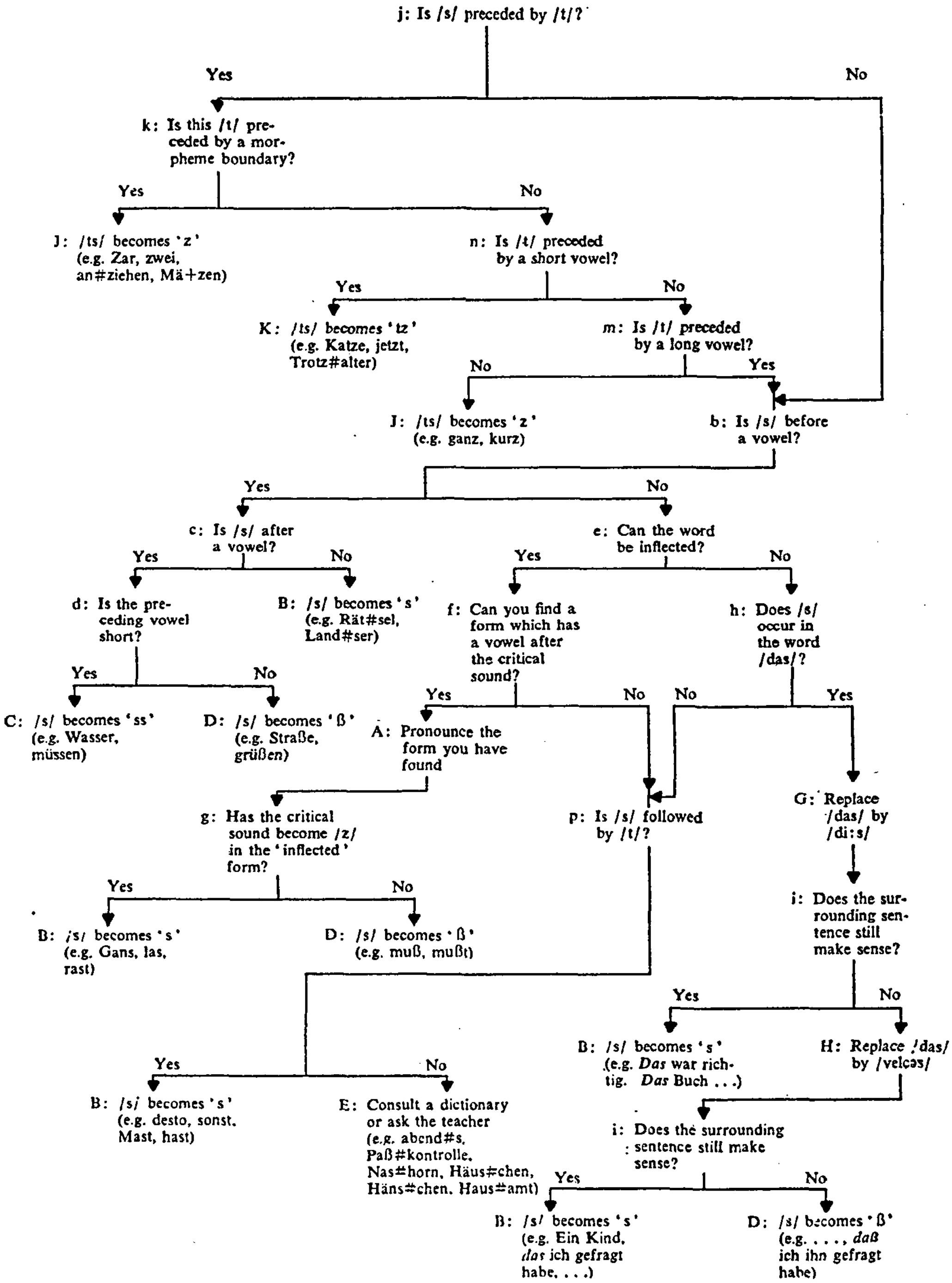
A quasi-algorithm (henceforth QA) establishes unambiguous associations between input and output words. We therefore extend the concept of QA and apply it not only to rules of type 2, but also of type 1. Now we say that all skills taught in a foreign language course can be described by QAs.

A QA must be explicit and effective. It is explicit and effective if it consists only of elementary operations. The concept " elementary operation " is absolute in respect of algorithms but relative in respect of QAs. Elementary operations are those which the addressee can execute expertly. A task is executed expertly if it is executed with the tools, the speed and the accuracy of an expert. It is not essential for the general definition of " expert behaviour " whether a task is executed unconsciously or consciously in accordance with a memorised procedure. The accuracy of an expert is never 100 per cent. Each specification of a task must allow a certain amount of tolerance, however small. The expert in speaking a foreign language is the native speaker of that language. If a pupil can find the spelling of the sound /s/ in a given word only by referring to a printed version of (1), then this operation is not yet subjectively elementary, however fast and accurately the pupil may perform the task, for (1) is not one of the normal tools of the expert, the native speaker. If a pupil can perform this task without reference to a written version of (1) but needs too much time or makes too many mistakes, then for him too this operation is not yet subjectively elementary. If a pupil is expert at all the constituent operations of a given QA (*i.e.* if he has the pre-requisite prior skills), then the skill of the pupil in the performance of the total skill described by the QA can be developed by means of practice in such a way that it gradually becomes an elementary skill. This practice is provided for the pupil by developing for each QA a set of exercises which the pupil has to carry out until he has mastered the total skill (complex skill) in question so that the complex skill has become an elementary skill for this pupil. Once this has happened the skill described by the QA can be used as an elementary operation (skill) in a later QA.

Before the pupil can apply a given QA and execute the associated exercises, he must have certain prior skills. These prior skills can be roughly determined by a fairly superficial inspection of any given QA and can be more precisely determined by the methods of prior knowledge analysis (*cf.* 1969c). For instance, (1) presupposes, *inter alia*, the following items of prior knowledge:

—Knowledge of the terms: morpheme boundary, vowels, long vowel, etc.
—Knowledge of the inflectional endings of certain words and of their pronunciation.
—Ability to determine whether a certain sentence makes sense or not, etc.

Each fact and each skill which are found to be pre-requisite, can in turn be described by a rule and thus by a QA. Each of these QAs is in turn associated with a list of prior skills and of QAs associated with these skills until we have



(1) Quasi-algorithm: Spelling of German /s/ (#represents a morpheme-boundary, + a pseudo morpheme-boundary)

reached a QA which does not demand any prior skills except those which the pupil possesses at the start of instruction. There are many such QAs. To find them is one of the most important tasks in the construction of a pedagogical grammar.

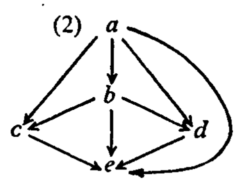
3. *Sequential relations and partially ordered sets*

If we have found that a skill *b* presupposes a mastery of a skill, *a*, we can represent this fact thus:

$$a \rightarrow b$$

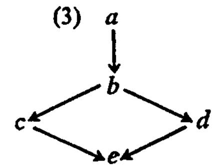
This means: *a* comes before *b*. It does not mean: *b* must come after *a*. Thus $a \rightarrow b$ indicates what one must *not* do but it does not prescribe what one must do. One may do anything except those things which are expressly forbidden. The sequential relations between pairs of QAs can be described by means of the $a \rightarrow b$ notation. There are many pairs between which such a relationship exists, but there are also other pairs between which there holds no such relationship in either direction. If the representations of these bilateral relationships are combined in one graph, then a structure such as (2) can result.

(2)



This graph can be interpreted as a partially ordered set (henceforth POS). The sequential relations of its elements follow automatically from the definition of the relationship $a \rightarrow b$ as "*a* must precede *b*." (2) can be simplified by omitting some of the arrows. This does not produce any change in the sequential relations between the elements of (2). The result of the simplification is (3), which we will also interpret as a POS.

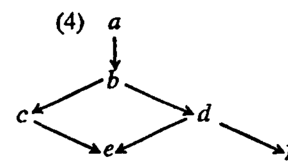
(3)



(3) says, for instance: *a* precedes *b*, *b* precedes *c*, *b* precedes *d*, *c* precedes *e*, *d* precedes *e*. Since *b* precedes *d* and *a* precedes *b*, (3) also says *implicitly*—just as (2): *a* precedes *d*, etc.

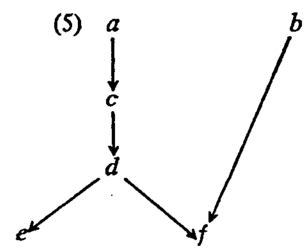
(3) has only one exit, *e*; we shall therefore call (3) a single-exit POS. We expect that prior knowledge analysis for programmed language instruction will, as a rule, produce multi-exit POSs, as illustrated in (4), where we have the two exits *e* and *f*.

(4)



The graphs (2), (3) and (4) had only one entrance (or "head"), *a*. We shall therefore call them "one-headed POSs." But one can also conceive of "multi-headed POSs" in foreign language instruction, *e.g.* (5)

(5)



Assume the nodes in (3) represent teaching steps (frames) and the arrows indicate the sequential relations between the steps.

Then there are two linear sequences by means of which this POS, (3), can be realised. The teaching steps are either worked through in the sequence $a+b+c+d+e$ or in the sequence $a+b+d+c+e$. Under no circumstances may they be worked through in the sequence $a+b+d+e$ $(+c)$ since this sequence does not satisfy the condition $c \rightarrow e$ (c precedes e). The arrows thus do not show alternative paths. After the pupil has worked through a, he must work through b, as there is no other alternative after a. Thus there is, between a and b, a positive sequential relationship, a fixed order. But when the pupil has worked through b, he may choose freely whether to work through c or d first. Thus after b there is no positive sequential relationship. But before the pupil works through e, he must have worked through both c and d.

4. *The pedagogical grammar*

We return to the QAs. QAs may contain only elementary operations. Non-elementary operations in a given QA, QA_j, can be made elementary if another QA, QA_i, together with its associated exercises, is placed before QA_j, and if QA_i describes the non-elementary operation contained in QA_j. Thus chains of QAs will originate. These can either be arranged in a linear sequence which satisfies the condition that all QAs must only contain elementary operations, or they can be assembled in structures of type (4), which we interpret as POSs and which also satisfy the condition that QAs may contain only elementary operations. The linear sequencing minimises the freedom of the consumer (pupil or teacher), whereas the POS maximises this freedom with losing anything in respect of the effectiveness of the teaching strategy. In programmed language instruction we therefore regard the partially ordered structure, the partially ordered program (POP) as more desirable than the linear teaching program.

We can now impose the following condition on the operations of a QA:

> (6) When a specific quasi-algorithm Q contains an operation OP, then OP must either be part of the initial skills of the addressee or Q must be preceded by a quasi-algorithm which describes the execution of OP in explicit terms.

We are now ready to define the concept of " pedagogical grammar " as it is used in the AALP-theory:

> (7) The pedagogical grammar is the finite, partially ordered set of the problems (tasks, skills) which occur in the manipulation of the mechanics of the target language. Associated with each problem is, as part of the pedagogical grammar, an explicit description of the problem-solving procedure (a quasi-algorithm). The most important feature of the pedagogical grammar is that its constituent problems (QAs) are (partially) ordered. They have to be ordered in such a way that a higher order problem does not contain any constituent problem which has not appeared in the pedagogical grammar before the higher order problem. Thus the problems in the pedagogical grammar are arranged in such a way that the pupil can learn the solution of the problems (the skills corresponding to each problem) in the sequence in which the problems occur in the pedagogical grammar.

For each QA, the teaching program

contains a number of exercises which have the effect of making the operation represented by the QA into an elementary operation. The exercises consist of teaching steps (in a AALP-program we usually have 10 teaching steps per exercise) and the teaching steps consist of teaching-step elements.

5. *Constructing the hierarchy of partially ordered sets*

We shall now try to determine the extent of the consumer's freedom in a partially ordered program (POP) by constructing a hierarchy of POSs, starting with the teaching-step elements.

(8) $T_i = [(M), (CP), QN, P, MR, (RR)]$

This means: A teaching step T_i is an ordered set of the elements M (message), CP (context prompt), QN (question), P (pause), MR (model response), RR (pause for repetition of model response). The elements appearing in parentheses can be replaced by the empty set, *i.e.* one or several of these elements may not be present. A " message " is a piece of factual information for the pupil; *e.g.* " In a German subordinate clause, the finite verb appears in the last place." An example for a " context prompt " can be found in the two teaching steps of (9):

(9) CONTEXT PROMPT	QUESTION	RESPONSE
1. Der Junge im Fluß schrie um Hilfe. Da sprang Herr Braun ins Wasser.	Warum sprang Herr Braun ins Wasser?	Weil der Junge um Hilfe schrie.
2. Frau Stör kam ins Gefängnis. Sie hatte ihren Mann umgebracht.	Warum kam Frau Stör ins Gefängnis?	Weil sie ihren Mann umgebracht hatte.

However, the context prompt can sometimes be given visually (*e.g.* as a drawing). In this case CP can start simultaneously with the question and can be continued until the end of RR.

QN can be coached in the grammatical form of a question, as in (9), or QN can appear in other guises by means of which the pupil is induced to make a response. For instance, QN can be merged with the message M, as for instance in a Skinner program (gap program):

(10) M + QN: In a German subordinate clause, the finite verb always occupies the —— place.

MR last

In (10) the gap in the message represents QN. The pause P is given implicitly as in all book programs where the pupil determines the duration of the pause.

(8) is called " ordered " in spite of the fact that some of its elements can overlap in respect of their temporal sequence in order to specify for AALP-programs that no element of (8) may appear before a prior element in the set. Thus M and QN are only merged when CP has been replaced by the empty set: QN never appears before CP, and MR never before P, etc.

We now combine the teaching steps into exercises, usually consisting of exactly 10 teaching steps each:

$$(11)\ E_i = \left\{ T_1, T_2, T_3 \ldots T_n \right\}$$

E_i is the set of teaching steps which are contained in an exercise We specify that, in an AALP-program, this set is either ordered or " totally unordered " (*i.e.* there are no sequential relations between its elements, all elements are freely interchangeable). If a specific exercise is " totally unordered " and if it is, moreover, presented by means of a tape-recorder, then the sequence in which the teaching steps are presented to any given pupil is adaptively determined by the RU-learning-algorithm (fully specified in 1967, Vol. 2, Chap. 1). We specify further that an exercise in an AALP-program is not to be interpreted as a POS even though this is at times theoretically possible, because the exercises are so short that treatment as a POS does not noticeably increase the freedom of the consumer whereas it has to be paid for by increased complexity of the resulting program. We therefore demand that when an exercise cannot be treated as " totally unordered " because of its internal structure, it is to be treated as strictly ordered. (9) is an example for the beginning of a " totally unordered " exercise, where it does not matter whether teaching step 1 precedes 2 or 2 precedes 1. An example for a strictly ordered exercise is (12):

(12) QUESTION	ANSWER
1. The old peasant woman was at the end of —— tether.	her
2. —— was trying to take this pig to the neighbouring farm, but the pig would not go.	she
3. —— turned left and —— turned right but —— would not walk in a straight line.	it it it
4. When old Mr. Brown had sent —— wife on this errand, —— did not know that this was a case where only Skinner ——self could help.	his he him

This exercise becomes meaningless and impossible to perform if the teaching steps are not worked through in the given order 1, 2, 3, 4.

We now combine all exercises which realise a given QA in a set Q_i. As a rule, Q_i will be a multi-exit POS:

$$(13)\quad Q_i = \left\{ E_1, E_2, E_3 \ldots E_n \right\}$$

Every set Q_i is called an " assembly " of exercises.

All assemblies Q_i of exercises which treat topics of a similar nature are now combined in a set S_i. Such similar topics are, for instance, all skills which are necessary for imitative articulation or all skills necessary for the production of sentence fragments. Correspondingly other sets S_i of assemblies Q_i can be formed, which deal with the production of whole sentences, with audio-comprehension, with reading comprehension, with writing in a foreign language, with the use of a dictionary, etc. S_i is a POS. We therefore say:

$$(14)\quad S_i = \left\{ Q_1, Q_2, Q_3 \ldots Q_n \right\}$$

S_i is the POS of the assemblies Q_i of exercises which treat topics of similar content. Each set S_i is called a " section."

We now combine all sections which operate in the same medium (*i.e.* which demand either acoustic or graphic responses) into the set C_i. We say:

$$(15)\ C_i = \left\{ S_1 \ldots S_n \right\}$$

C_i is a POS. Each set C_i is called a " component."

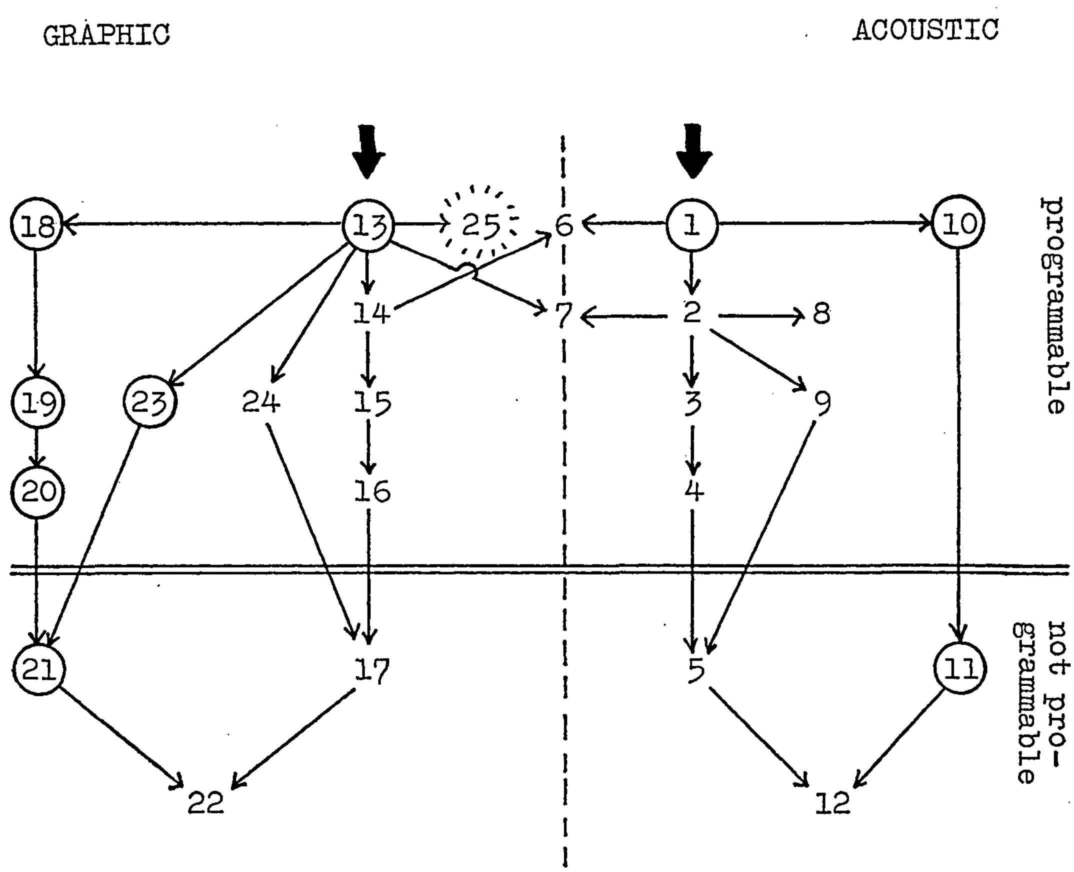
Finally we establish the set D_i of the components which together represent the communication skills (as in the Δ-Diagram shown below).

$$(16)\ D_i = \left\{ C_1, C_2 \right\}$$

D_i is " totally unordered," *i.e.* C_1 and C_2 do not presuppose each other and they can be worked through in any sequence.

(17) shows the so-called Δ-Diagram, which represents the set D_i and its sub-

The Δ-Diagram
Communication skills



(For explanation, see overleaf.)

Explanation of (17)

1 Sound discrimination
2 Imitative articulation
3 Sentence fragments
4 Sentence structures
5 Free speaking
6 Dictation *
7 Reading out aloud *
8 Fixed phrases
9 Vocabulary and idioms
10 Aided auditory comprehension
11 Free comprehension
12 Conversation
13 Sign recognition
14 Sign reproduction
15 Aided writing
16 Production reference grammar
17 Free writing
18 Systematic guessing
19 Graphic comprehension
20 Recognition reference grammar
21 Free reading
22 Correspondence
23 Recognition dictionary
24 Production dictionary
25 Formal grammar

The two thick arrows denote entry points.
* These are connective skills, not communication skills.

sets C_1 (graphic component) and C_2 (acoustic component) and sections 6 and 7 (dictation and reading out aloud), which have been inserted in order to connect the two components. (A discussion and justification of the content of the $\triangle$-Diagram and of its arrangement, together with proposals concerning programming techniques and media of presentation for the various sections can be found in 1967, Vol. 2, Chap. 6).

The discussion of sequential relationships and the definition of $a \rightarrow b$ as "a precedes b" indicates clearly the meaning of the arrows in the $\triangle$-Diagram. It is, moreover, obvious by now that the individual sections in the $\triangle$-Diagram represent hierarchies of partially ordered sets.

Part 2 (Vol 8, p 122-124)

THE CONCEPT OF PARTIAL ORDER IN LANGUAGE PROGRAMMING AND THE FREEDOM OF THE CONSUMER: PART 2

KLAUS BUNG
Department of Linguistics, Cambridge University

THE FREEDOM OF THE CONSUMER

THE term " consumer " in this paper denotes pupils and teachers. We are trying to offer the consumer as many choices as possible within the teaching topic and among various possible teaching methods and to impose as few restrictions as possible but as many restrictions as necessary for sound teaching (*i.e.* sound sequential relationships). In making choices, the pupil has priority over the teacher. Pedagogically undesirable choices are excluded *a priori* by the program and they will thus never be offered to the pupil. The pupil is free to choose among the alternatives which are still open within the limits set by sound teaching principles, and he will make this choice according to his tastes and his learning aims. Only in those cases where the pupil does not wish to exercise his right of making choices, the teacher in charge of programmed instruction is entitled to make the final choice on behalf of the pupil.

Let us assume the sets T_i, E_i, Q_i and S_i were strictly ordered and only C_i and D_i were not strictly ordered, where D_i is to be " totally unordered " and we impose the partial order represented in the Δ-Diagram on C_1 and C_2. This would be the least favourable case for the freedom of the consumer in a programme which has partial order at least at the highest (coarsest) level of resolution. Even in this least favourable case, the fact that C_i is a POS suffices to give the consumer the following choices:

Once he has worked through section 1 (S1), he may turn to S2, or S10 or S13. Once he has worked through S1 and S2, he may turn to S10, 8, 9, 3, 13. Let us present some of these choices in a table:

(18)

After working through sections . . .	the consumer can choose from among sections . . .
1	2, 10, 13
1, 13	2, 10, 14, 18, 23, 24, 25
1, 2, 13	10, 14, 18, 23, 24, 25, 7, 3, 9, 8
1, 2, 13, 14	10, 18, 23, 24, 25, 7, 3, 9, 8, 6, 15
etc.	

The more sections have been worked through, the more choices become open. In the above discussion of choices, we have assumed that the sections are indivisible. If, instead, we had assumed that each section of the $\triangle$-Diagram represented a hierarchy of *single-exit* or *single-headed* POSs, we would have obtained the same limiting effect in respect of the freedom of the consumer. If both a and b are either indivisible or represent hierarchies of *single-exit* or *single-headed* POSs, then $a \twoheadrightarrow b$ must be interpreted in such a way, that b must not be started unless all subsets of a have previously been completely worked through.

However, there is no reason why we should expect that the POSs under discussion here are single-exit or single-headed POSs. Let us assume realistically that Q_i, S_i, and C_i are *multi-exit* and *multi-headed* POSs, and that in case of $a \twoheadrightarrow b$, both a and b are not only *multi-exit* and *multi-headed* POSs but even hierarchies of *multi-exit* and *multi-headed* POSs. This assumption describes realistically the facts underlying the $\triangle$-Diagram. Its consequence is the following interpretation for a and b in $a \twoheadrightarrow b$, and the same applies to all sections of the $\triangle$-Diagram connected by arrows:

(a) b must not be started unless at least one exit of a has been reached.

(b) Only when *all* exits of a have been reached, *all* parts of b become accessible.

This interpretation of the $\triangle$-Diagram increases the choices (*i.e.* the freedom) of pupil and teacher tremendously, as now there are choices not only within the elements of C_i but also within the elements of the sub-sets represented by C_i and within the sub-sets represented by these sub-sets in turn.

This great amount of freedom, however, is not paid for by producing methodological chaos, it is not equivalent to introducing a " free-for-all " into teaching methods. For the system indicates precisely the limits of the freedom given to the teacher. If he ignored these limits, he would make methodological mistakes. Thus the language program, conceived as a partially-ordered set of elements, maximises the freedom of the teacher without impairing his didactic efficiency.

By contrast, the conventional linear and branching program obviously minimises the freedom of the consumer by choosing more or less arbitrarily only one of the many sequences of teaching steps permitted by the partial order. This is true even of branching programs because they too are based on a strictly linear sequence of teaching steps, sometimes called the " fastest path," *i.e.* the path taken by the addressee who makes no mistakes. Neither the pupil nor the teacher has any choices concerning the branches because these are strictly controlled by the pupil's errors. The freedom inherent in the structure of the subject matter is usually not exploited.

It could be objected that the POP is superior to the classical types of teaching program in respect of the freedom of the consumer but not to the conventional live teaching in the classroom, in which the teacher can, within certain wide limits, do whatever he likes, provided his teaching is reasonably successful. We therefore have to ask: Is not POP an improvement of something that is implicitly, *a priori*, bad, but an improvement which will never reach the degree of freedom of the original, of traditional classroom teaching?

Both traditional classroom teaching and POP are subject to the same restrictions, namely that the procedure of the teacher or the program must not offend against the laws inherent in the structure of the teaching topic, laws which can be discovered by the techniques of prior knowledge analysis. These laws are valid no matter whether the teacher is conscious of them or not. It can happen that the teacher teaches intuitively or accidentally in a sequence which is methodologically permitted. As a rule he teaches consciously in a specific sequence of which he knows by experience that it leads to success. About these successes two observations can be made:

(a) Even if the chosen sequence (or the sequence inherited from the teacher training college or from a textbook) finally leads to success for many pupils, this sequence could still contain wrongly constructed subsequences which make success more difficult, delay it or, for some pupils, prevent it.

(b) Although the teacher knows the positive sequence which he uses, he does not know the other possibilities which are open to him.

As he does not know the permitted possibilities, he cannot fully utilise his theoretical freedom without running the risk of making methodological mistakes. He is like a man in a dark wood, who regularly goes home on a certain road, but never tries to find himself a new route through the wood, because he does not know where the abysses are. His freedom to walk through the wood anywhere he likes is thus purely theoretical. In practice, the darkness minimises his freedom of movement and he is like Goethe's beast on a barren heath, led in a circle by an evil spirit while all around are beautiful green meadows.

The darkness in our metaphor represents, in the teaching situation, the teacher's ignorance about the question of which pedagogical decisions have negative consequences. A partially-ordered program removes this ignorance, not by telling the teacher which teaching steps he *must* perform, and not by telling the teacher positively what would be the consequences of certain misguided pedagogical decisions. (In the absence of wrong decisions, knowledge of their consequences is of no interest.) POP confines itself to telling the teacher explicitly *what he must not do.* The general rule is: Everything is permitted, except that which is expressly forbidden. All these negative specifications can be reduced to the form $a \rightarrow b$, where $a \rightarrow b$ means either " b must not be treated before a " or " b must not be started unless at least one exit of a has been reached, and b may not be treated completely unless all exits of a have been reached."

These basic specifications of partially-ordered programs sound very sober and modest, and so they are. But, as we have shown, these modest specifications, if consistently developed, have a tremendous effect on the freedom of the pupil and the teacher. Thus we can add a new argument to the general arguments in favour of programmed instruction:

A teaching program (if constructed as a POP) is not only in many cases more effective than live-teaching, but it also maximises the freedom of the consumer, whereas this freedom is minimised by traditional live-teaching and by classical teaching programs.

If, one day, sufficient effort and money is invested into basic research about programmed language instruction, this argument might become decisive in deciding whether or not programmed language instruction becomes the generally accepted form of language instruction.

REFERENCES

BANERJI, R. B. (1969) *Theory of problem solving. An approach to artificial intelligence.* New York: American Elsevier Publishing Co.

BUNG, KLAUS (1967 and 1968) *Programmed learning and the language laboratory*, 2 Vols. Longmac, Research Publications Services, 11 Nelson Road, London S.E.10.

BUNG, KLAUS (1969a) "Towards truly programmed language laboratory courses," *Audio-Visual Language Journal* **7**, 5–17. Audio-Visual Language Association, London.

BUNG, KLAUS (1969b) "The function of the teacher in programmed language instruction," *Recall: Review of Educational Cybernetics and Applied Linguistics* **1**, 55–78. Longmac, Research Publications Services, 11 Nelson Road, London S.E.10.

BUNG, KLAUS (1969c) "Prior knowledge analysis for programmed language instruction." *Recall: Review of Educational Cybernetics and Applied Linguistics* **1**, 98–117. Longmac, London.

BUNG, KLAUS (1970a) *Ansätze zu einer Theorie des programmierten Sprachunterrichts* (Towards a theory of programmed language instruction). To be published by Julius Groos Verlag, Heidelberg (English translation to be published by Mouton, The Hague).

BUNG, KLAUS (1970b) "Programmed language teaching and its implications for teacher training." In: R. C. Lugton (ed.): *Preparing the EFL teacher: A projection for the 70s*, 119–177. Philadelphia, Pa., U.S.A.: Center for Curriculum Development, Chilton Books.

CARROLL, JOHN B. (1963) "A primer of programmed instruction in foreign language teaching." In: IRAL, Vol. 1, No. 2, 115–141. Heidelberg: Julius Groos Verlag.

CHOMSKY, N. (1959) Review of *Verbal Behavior* by B. F. Skinner. In: *Language* **35**, 26–58.

FRANK, HELMAR (1969) *Kybernetische Grundlagen der Pädagogik* (Cybernetic foundations of educational theory). Second edition. 2 Vols. Baden-Baden, Germany: Agis-Verlag.

GOODSTEIN, R. L. (1963) *Boolean algebra.* Oxford: Pergamon Press.

GLUSHKOV, V. M. (1966) *Introduction to cybernetics.* New York: Academic Press.

HEIMANN, P. (1962) " Didaktik als Theorie und Lehre " (The theory, and the teaching, of didactics), *Die Deutsche Schule* 407–427.

LANDA, L. N. (1968) " Algorithms and programmed instruction." In: Bung (ed.) 1968, pp. 57–135.

LANDA, L. N. (1969) *Algorithmierung im Unterricht* (Algorithmisation in the teaching process). Berlin: Verlag Volk und Wissen.

LÁNSKÝ, MILOŠ (1969) "Learning algorithms as a teaching aid," *Recall: Review of Educational Cybernetics and Applied Linguistics* **1**, No. 3, 81–98.

MARTY, F. (1962) *Programming a basic foreign language course. Prospects for self-instruction.* Roanoke, Virgina: Audio-Visual Publications.

MORTON, RAND (1959) *The language laboratory as a teaching machine.* Ann Arbor, Michigan.

MUELLER, T. H. (ed.) (1968) *Proceedings of the seminar on programmed learning (i.e.* language learning). New York: Appleton-Century-Crofts.

ORE, O. (1963) *Graphs and their uses.* New York: Random House.

SKINNER, B. F. (1950) " Are theories of learning necessary? " *Psychological Review* **57,** 193–216.

SKINNER, B. F. (1957) *Verbal behavior.* New York: Appleton-Century-Crofts.

SPOLSKY, BERNARD (1968 " A psycholinguistic critique of programmed foreign language instruction." In: Bung (ed.) 1968, pp. 155–167.

STOLL, R. R. (1961) *Sets, logic and axiomatic theories.* London.

VALDMAN, A. (1966) " Programmed instruction and foreign language teaching." In: A Valdman (ed.): *Trends in language teaching.* New York: McGraw-Hill.

[*To be continued*]