

# ソ 連 の C A I

## ——語学教育の状況——

### ОБУЧЕНИЕ С ИСПОЛЬЗОВАНИЕМ ЭВМ В СОВЕТСКОМ СОЮЗЕ ——МЕТОДИЧЕСКИЕ ОБСТАНОВКИ ПРИ ОБУЧЕНИИ РУССКОМУ ЯЗЫКУ——

山 田 隆

80年代にCAIは終わったとささやかれた時期があった。パソコンの普及にもかかわらず、画一的な学習に見切りをつけた訣別状だった。キーボードの他に、強力で魅力的な入出力機器を持たなかったことにも一因はあった。コンピュータを使用する教授法上の混迷もあった。時間だけがだらだらと過ぎた。

CAIは終わったといわれながら、今なお学習ソフトが開発される。ところで、CAIとは何か。Computer Aided Instruction. もしくは、Computer Assisted Learning. ロシア語訳では Обучение с использованием ЭВМ. ここからコンピュータが終始学習の主導権を握り、すべての学習課程を切り盛りしていくとは考えようにもできないし、且つ期待できない。これを承知の上で、CAIプログラムを稼働し、学習していても、いつの間にか自分がコンピュータの指図通りに動いていることに気付くであろう。学習ソフトのスイッチを入れれば閉塞的状况におかれるコンピュータと学習者の関係は、主従的なそれである。コンピュータの指図や進め方が気に入らなければ、電源を落とすより他に打つべき手がない。そのような性格を持っている。

古典的な実験だが、人間の代りをコンピュータが勤め得るかの検証基準の一つとしてこのようなものがある。互いに隔離された状態で、キーボードを使用した人間同士の交流から一方を、他方への予告無しに機械に切り替えて、以前と変わりなく交流を持続できるなら、その限りで機械が人間の役割を果たしてい

るというものである。

判定用具を「キーボード」に限定するのは現在の機器進歩の実態を反映していないとの反論があろう。これは認める。ユーザをどの程度引きつけ、且つ持続的な学習を可能にするかの観点からコースウェアを評価する時、ハードウェアに関してはマルチメディア化の程度に比例して効果が期待できる。

- 1) テキスト画面とビープ音のみ (例えば "HEBREW" on IBM-PC) よりは
- 2) グラフィック併用の方がよい。
- 3) それより外部周辺機器 (ステレオ, CD, ビデオなど) を持続した方が遙かによい。

事実、パソコン上で稼働する最近の語学コースウェアは、必要とする環境がますます肥大している。メモリーや演算速度ばかりではない。必須機器に CD プレーヤを含めるものさえある「ASA hI パソコン」の 1991 年 8 月 1 日号の特集記事「今度こそマルチメディアでペラペラに・ラクして学ぶ、パソコンで英語」のバラエティを参照していただきたい。しかし、キーボードが他の周辺機器に置き換わろうとも、何かが付加されようとも、根本の性格が変わるであろうか。

コンピュータによる学習システムを評価するに際して、2つの評価基準に大別される。

1. ハードウェアの構成如何による評価
2. ソフトウェアのアルゴリズム如何による評価

コースウェア作成にはハードウェアの限界が全面的に働いている。これがコンピュータ学習の評価に良くも悪しくも影響している。演算速度が向上し、付属機器が多彩になるほど、利用の可能性が大きくなる。

一方、ハードウェアの進歩とは別に、ソフトウェアの質的良否を問うことにも意味がある。この際の評価対象として、一つには、既に開発されているコースウェアセットを評価すること、他方に、理論として提唱されているアルゴリズムを検討することが可能である。後者の選択は、所与のコースウェアの入手が困難もしくは未完成の段階にあるときに有効である。ソ連で開発されるソフトウェアの大半がこのグループに属する。

従来から CAI には 3 つの方法があるとされている。スキナーが行動主義心理学からヒントを得たというドリル式に始まり、ヒューリスティックな方式、それとシュミレーション方式がそれだ。

ドリル式の欠陥を指摘され、CAI は終わったとさえ思われた。にもかかわらず、ドリル式のコースウェアは、依然と世界を席卷している。ソビエトとて例外ではない。

CAI の歴史を 7 つの段階に分類した論文が米国で発表されたが、その変遷段階を遅ればせながらわが国でも辿っていると述べる。ハードウェアの普及具合から推し量れば、これはそのままソビエトにも当てはめることができるのかもしれない。

その仮説を紹介しよう。

「日本はアメリカの轍を踏むのか」\*と題して

#### 第 1 段階 校務・教務での利用 (1950 年代)

授業料、給与計算、時間割り作成

#### 2 CAI (1967 年の PLATO の開発から 1970 年代)

- ・理論的背景は行動主義心理学であった
- ・学習者の「積極的役割」に注目する理論の台頭によりダメージをうける
- ・コンピュータの分散化により見込み違いを引き起こす (マイクロコンピュータの台頭)

#### 3 マイクロコンピュータの登場 (PLATO 開発にやや遅れた時期)

- ・オーサリング言語の開発と手軽にコースウェアを作成できる時代になる
- ・しかし、コンピュータ失望の時代でもあった

---

\*『ハイパーメディアと教育革命』、浜野保樹著、アスキー出版局、1990 年、pp. 45-53.

- 4 LOGO の思考の道具としてのコンピュータ (1980 年「マインドストーム」出版)
  - ・ LOGO は学習の哲学を含んでいた
  - ・ しながら教育現場に目的も定まらないままハードウェアを大量導入した時期でもあった
- 5 コンピュータリテラシー運動 (1980 年代初頭から)
  - ・ 「コンピュータ教育」科目を設置
    1. コンピュータフィットネス
    2. プログラミングを目指す
- 6 道具としてのコンピュータ
  - ・ ワープロは書くために、データベースは歴史や動物の分類のためにという通念通念が拡大浸透する
  - ・ プロダクティビティ・ツール (ワープロなどの教科と関係なく利用可能なソフト) が普及する
- 7 主要教科の中へのコンピュータの統合
  - ・ ソフトウェアの備品化・教材の一部として取り扱われた

そこで、ソビエトにおけるコースウェアの教授法に的をしぼり、どちらかという、事例紹介や教授法理論の紹介を行いたいと思う。

これは青少年向けに書かれた『サイバネティックス 始めから終わりまで』(1990) という百科全書\*の「学習マシン」の項目で紹介された事例である。

最初の事例のシステム名称は「女子生徒」(УЧЕНИЦА)。適用科目として、物理、化学、生物その他が考えられ、プログラムに応じて

1. 蒸気機関の動作モデル
2. 降雨・降雪

---

\* “КИБЕРНЕТИКА от А до Я”, Виктор “Детская литература”, 1990, Москва, стр. 255-256.

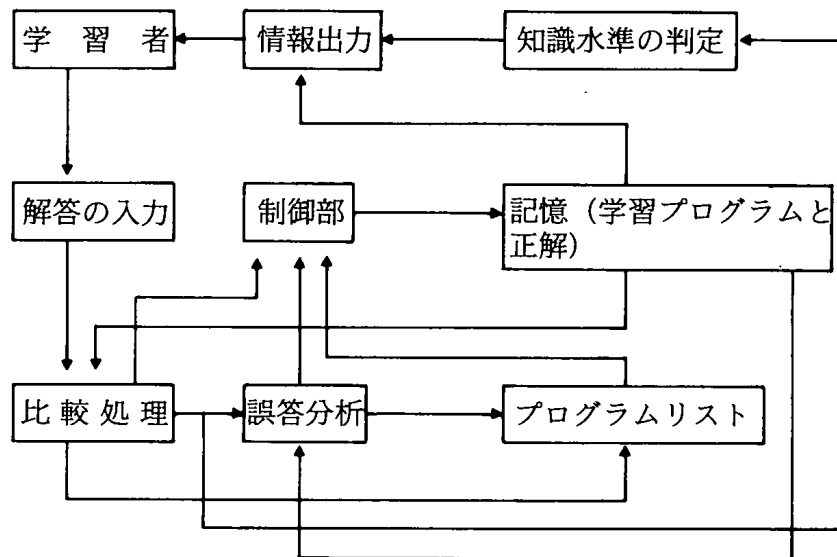
3. 溶液の化学反的的色彩変化

4. 植物根による食物吸収過程などをディスプレイ上に出力する。

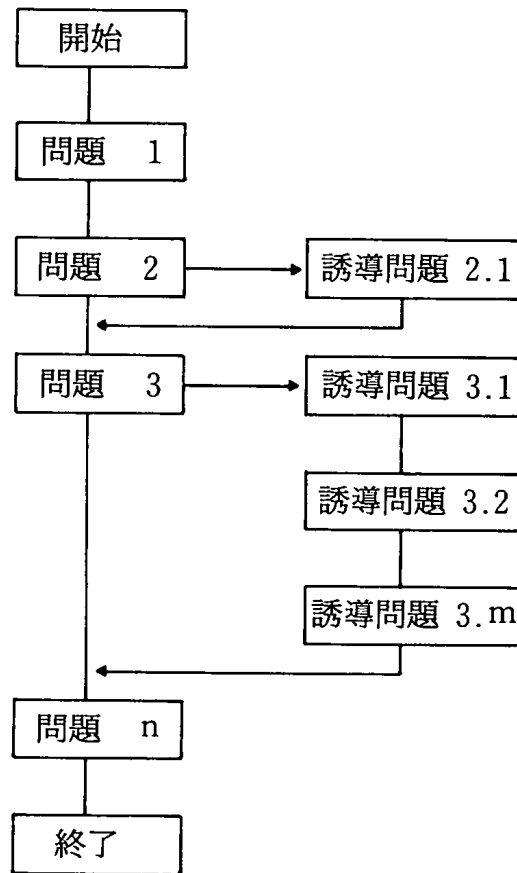
また、教師と生徒間の関係は、任意の生徒のディスプレイにいつでも「接続」できることを接点としている。

これは、また、いわゆる分岐型プログラムによる学習システムである。学習者が正解を見いだせない場合、枝分かれ部を通過しなければならないのが、分岐型の特徴である。

「女子生徒」システムの構造図



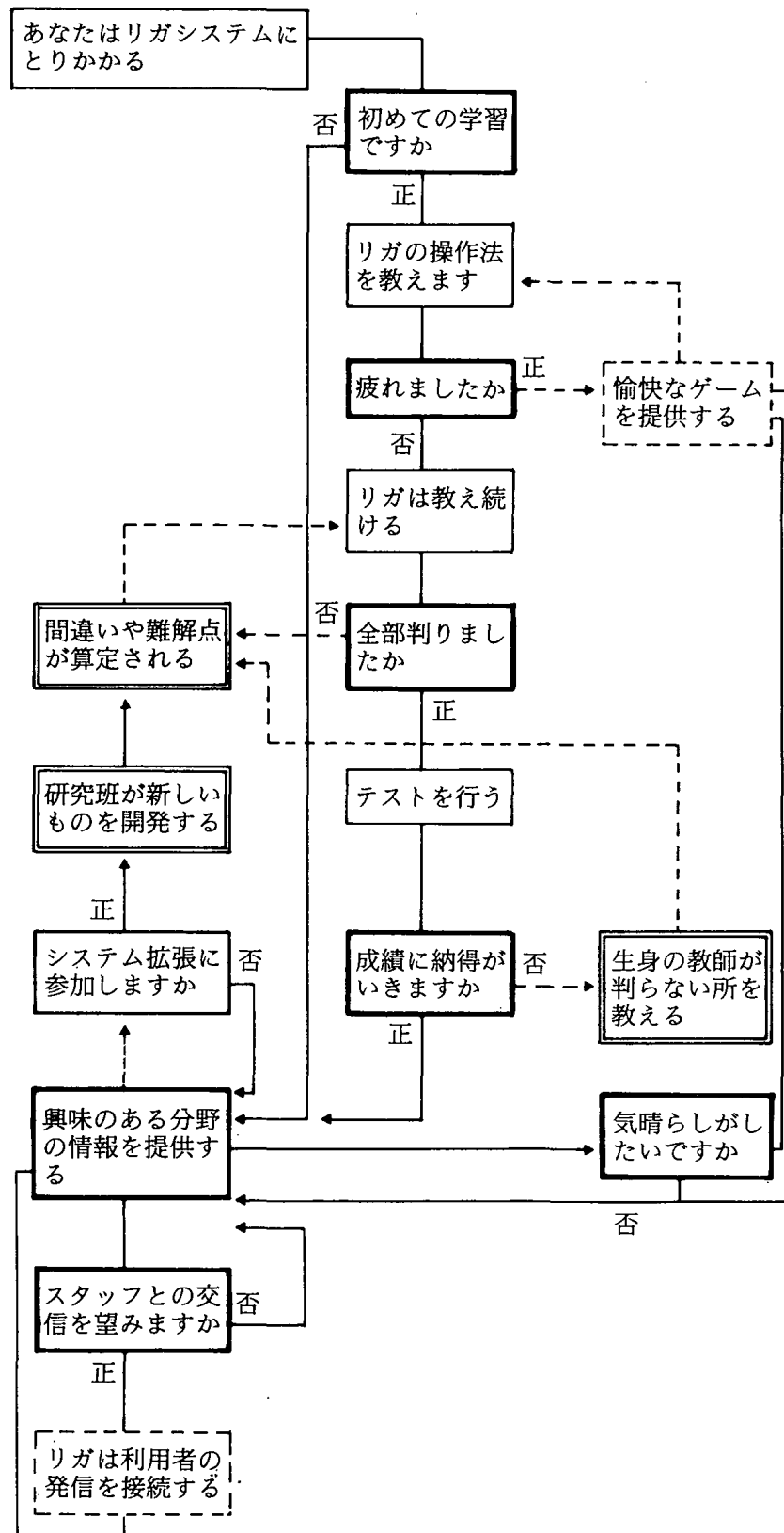
「女子生徒」システムの出題と解答処理の方法



2 番目の「リガ」システム(РИГА)は, 大学生用 CAI システムで, コンピュータルームでのような集団利用を前提としている。コンピュータが教師に「成り代わり」,

1. 教科書
2. ワークブック
3. 一目瞭然のイラスト教材を提示する。
4. 愉快で, のめり込むようなゲームのパートナーを勤める
5. 電子メール, 電子ボード, インフォメーションセンターを果たす

「リガ」学習システムのあらまし



本システムは構造図を見る限り、「興味ある分野の情報を提供する」以降の終結ルーチンがない。ここが不明である。

「コルベット」(КОРБЕТ) システムは、小中学校、工業高校、職業訓練校、そして大学での使用を目的とする。コルベットは、プログラミング、物理、数学、化学、ロシア語、文学、図画の基礎を教える。64 KB のメモリーをもち、1 秒あたり 62 万 5 千回の演算をこなす。

もう 1 つ実例を紹介しよう。「外国におけるロシア語」誌 (1990. 12)\* に掲載された事例だが、要旨を以下に記す。

#### コンピュータソフトウェア「ロシア語 一步一步」

米ソ合併企業『パラグラフ (ПараГраф)』社は、コンピュータソフトウェア『一步一步のロシア語』を提供している。米語版は『Russian Step by Step』、仏語版は『Le Russe pas a pas』である。著者は、Е. И. Василенко, Э. С. Ламм, А. М. Якубович, А. Ю. Демидов である。

本コースウェアは、学習用、および試験用エキスパートシステムの原理に従って作成された。基礎にしているのは、ワシレンコ・ラム共著の『ロシア語 独習』の教授法である。「ロシア語 一步一步」による学習の結果、約 1,500 語の単語と日常生活をテーマにするロシア人との会話モデルの用法を習得することができる。このためにコースウェアでは練習問題や学習用ゲーム、規則などの形で基本文法のコースが指示される。

このシステムは、以下の仮題をもつ、いくつかのモードの中で作動する。

1. 学習
2. 文法
3. 試験
4. (誤答の) 分析と統計

---

\* “Русский язык за рубежом”, 1990 —6.

5. (システム) オプション

6. 指導 (ヘルプ)

7. コース

「学習」モードは、システム内の基本プログラムである。ここで一步一步と語彙、文法、必要とされる発話構造を勉強することになる。充分手応えありと判断したら、「試験」モードの学課を試行するのもいいだろう。これら2つのモードは、試験、分析、誤答の訂正と説明、その他必要な規則や図式の表示をおこなう。また、「学習」モードは、専用レッスンによるロシア語キーボードを学ぶ可能性を提供している。

「文法」モードは、学習者の選択したテーマに沿った全ての文法規則、図式、一覧表を一覧し、かつ、所与のテーマを学習するために用意された訓練をおこなうことができる。

「分析」モードは、各々の文法テーマに即した学習で得られたデータの集約を保証する。課題を解くことを試みた回数の合計、正(誤)答数、正答率、解答作成時のシステムへの問い合わせ回数など、語答に対する分析と統計についてはコースウェア作動中のいかなる時点においても知ることができる。

「ゲーム」モードは、学習形態を変更し、しばし休憩する可能性を与えてくれる。

「オプション」モードは、システムパラメータの再定義を可能にする。

「ヘルプ」モードは、コンピュータコースウェアでの学習の基本モードとステップについての詳細な情報を提供する。

「コース」モードは、システムに組み込まれた練習問題を利用しつつ、担当教師の教授法に適合するように補助的な教材を構築する可能性を与えてくれる。

「ロシア語 一步一步」は、首尾一貫互いに関連をもつ、9つの単元から成っている。同時に、各々が独立で用いることができるようにも設計されている。各単元が10課から、また一課は4～8つの練習問題から成る。一課あたりの学習時間は、おおよそ30分である。

コースウェアでの学習には IBM-PC コンパチのコンピュータが必要で、MS

-DOS V 3.1 以上の OS のもとで作動する。コンピュータには EGA もしくは VGA 規格のカラー、またはモノクロモニターが付属していなければならない。作動させるための記憶容量は 512 KB である。「ロシア語 一步一步」は、5.25 または 3.5 インチフロッピーディスクで供給され、コピープロテクトがかけられている。

資料で紹介する「プログラムのタイプ、および教授法」の中で注目すべき考察がある。

- 1) プログラムをモジュール化して、共有財産にする構想
- 2) その具体化案として、コースウェア作成支援システムを提唱するの 2 点である。プログラムの生産性を向上させるための、共有財産を蓄積するための効果的な提案であり、両者とも既に実施に移されている事柄だ。

しかし、こうした方法だけではプログラムやデータベースの枠組みを超えることはないし、また、超えることができない。例えば、学習者の「反問」を吸収することは考えられないであろうか。ドリル方式では、教材が提示され、必要な説明が施され、さらに地均しの練習、次に自分の実力を確実なものにするための仕上げと一連の、既にプログラム化されたステップが順々と続くのである。ドリル式プログラムである限り、閉塞したコースからの逸脱は許容されない。コンピュータに敢えて質問してみたら、コンピュータは反応を示すだろうか。その時学習者に与える刺激はどんな結果になるであろうか。

また、コンピュータに例題のシュミレートをしてもらう訳にはいかないのだろうか。プログラムの中で「読む」の動詞が提示されたのであれば、コンピュータが学習者の質問に答えて「働く」の語尾変化をいつてみせるのである。そうならば学習者は「コップ」の格変化の正否をコンピュータに質すことも可能である。

#### 仮想例の 1

学習者「例題は理解できました。ところで、работать はどのように現在変化させるのですか」

### 仮想例の 2

学習者「例題は理解できました。ところで先日 танцуют という単語をみか  
けたのですが、これの不定形と他の現在変化形はどうなるのですか」

### 仮想例の 3

学習者「Копп стакан の格変化は, стакан, стакана, стакану, стакан, стакан-  
ом, стакане で間違いありませんか」

この場合、問題になるのが「マンマシンインターフェースの種類」と「コン  
ピュータの言語理解度」である。前者は、入出力方法の問題である。後者は、  
学習者が発する質問をコンピュータがどの程度理解し得るのかの問題である。  
ロシア語には多彩な語尾変化がある。この処理だけでも機械に大きな負荷がか  
かる。スペルチェック、構文分析などは大仕事のプロジェクトだ。参考資料と  
して巻末にスペルチェックプログラムのモニター記事を紹介した。

CAI は使命を終わっていない。CA までは誰にでも理解できるし、同意を得るこ  
ともできるだろう。その後の I は何を意味しているのか。インストラクション。  
語学学習にはインストラクションの要素が否定できない。語学は「経験的学問」  
ともいう。では、インストラクションにどのような意味を認めるのか。コン  
ピュータが指導のみならず、反問を容認して学習者を大いに啓発 (instructive)  
してくれることを願う。

### 参考資料その 1

CAI プログラムの分類と作成について。出典は、  
(библиотека преподавателя русского языка как иностранного)  
ЭВМ В ПРЕПОДАВАНИИ РУССКОГО ЯЗЫКА КАК ИНОСТРАНН-  
ОГО, С. В. ФАДЕЕВ, Москва, "Русский язык", 1990 である。

## CAI プログラムのタイプ

コンピュータ用プログラムの分類は、通常いくつかの基準に従って行われる。

1. 機能目的
2. プログラム作成法、もしくは構造
3. 学習者の特性に適合する能力
4. 構成要素配列の数と順次性
5. 教授法上の目的
6. プログラムの独立程度
7. テスト研究の複雑度

機能目的に応じて次のタイプのプログラムに分類することができる。

1. 学習用
2. 情報提供の
3. 試験の
4. 遊戯的な
5. 複合的なプログラム

「学習プログラム」とは通常学習上の情報（教材）と理解され、そこではマスターすべき知識、能力、習慣、および情報化の方法が記述される。学習情報は学習者の知識水準を引き上げることを目的とし、客観的に存在する伝達内容の認知される総体である。他言すれば、コンピュータプログラムの学習情報とは、コンピュータによるさまざまな伝達内容、すなわちテーマ、目的、指導、課題などと理解すべきである。学習プログラムの基本的目的は、一定の知識、能力、習慣の獲得のためにコンピュータと学習者の対話の提供である。

「情報提供のプログラム」の目的は、一定の情報バンクから任意の概念に関する対話の提供である。情報提供プログラムに基づいて言語資料に関する単一

のコンピュータバンクが作り出され、これに関連するデータベース

1. 音声学上の
2. 意味論上の
3. 文法上の
4. 総辞論のデータベース

が付加される。

「試験のプログラム」は、程度のあれこれの単位、テーマに関して学習者の知識と認定と矯正を目的とする。試験プログラムの基本的要素は課題、試験問題、解答のバリエーション、知識、能力、もしくは習慣の程度の試験を実現するためにその他必要とされる要素である。

「遊戯的プログラム」を、仮に、1. 純粋に遊戯的なプログラム、2. ロシア語資料による遊戯的プログラム、3. 状況モデル化した遊戯的プログラムに分類することができる。

純粋に遊戯的なプログラムは、通常、基本プログラムとは何等関係がなく、これらのプログラムは、弛緩期の一服、もしくは基本プログラムの課題を正確に解いたことに対する報償としてのみ基本プログラムの一部を成すにすぎない。このタイプの遊戯プログラムに分類されるのは、有名なゲームの“Крестики-нолики”(3 並べゲーム)“Морской бой”(海戦)、“Лабиринт”(迷路)、“Ханойская башня”(ハノイの塔)“Кубик Рубика”(立方体ゲーム)など多数ある。

ロシア語資料による遊戯プログラムは、通常、純粋に遊戯的なプログラムと同様の機能を果たす。とはいえ、もし後者が基本プログラムと何等関連性がなく、基本プログラムから除外することが容易なら、言語資料を用いた遊戯プログラムは、基本プログラムとより緊密な関連を持ち、学習者にロシア語に関する補足的な情報を与える、さもなくば、ロシア語のあれこれの語彙文法上の現象をより効果的に習得する一助になるであろう。

状況モデル化の遊戯プログラムは、外国語としてのロシア語学習においてももっとも将来性があり、単独でも、他のプログラムとの組み合わせにおいても用いられる。状況モデル化遊戯の助けをかりて、学習者はかなり複雑な語彙文

法上のテーマ，たとえば運動の動詞，動詞の体などに磨きをかけることができる。上級者レベルでの状況モデル化プログラムの利用は，学習者のさまざまな職業的習慣の形成を促進する。

機能目的別にコンピュータプログラムを分類することは，あらゆる分類と等しく極めて条件付きのことである。なぜなら，現実の学習においてはこれらのプログラムが組み合わされた形で機能し，情報，学習，試験，遊戯といういくつかの要素を含むからだ。具体的なコンピュータプログラムにおけるこれらの要素は，さまざまの分配と結合の中で供与される。たとえば，外国人がロシア語を学習する時，言語資料の単一バンクに含まれる情報提供のプログラムは，学習プログラムにとっては専門的な情報プログラム（学習付けの情報プログラム）として利用される。

学習プログラムは，通常，管理の要素も含む。学生の独習用プログラムでは大抵試験の要素が勝っている。このことがこの種のプログラムを管理学習プログラムに類別することが可能だ。

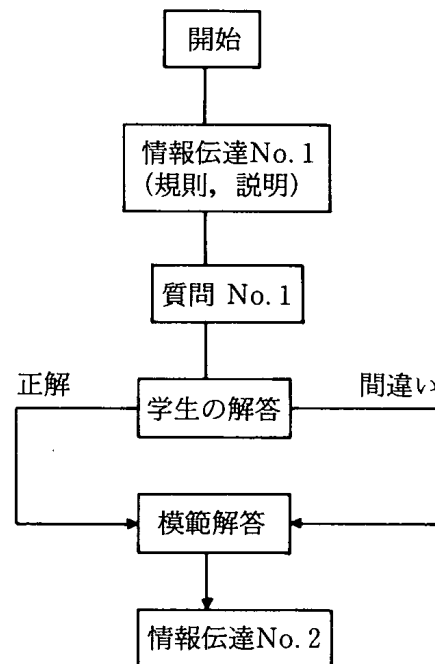
プログラム化の方法に関して，すべてのプログラムを3つのタイプに分類することができる。

1. 線条的
2. 分岐的
3. 融合的（組み合わされた，混成的な）プログラム

この分類の基礎には利用者がコンピュータプログラムの作成に際して実行しなければならない手順がおかれている。

線条的なプログラムとは，試験の課題や質問に対して学習者の解答如何に係わりなく唯一可能な作業方向を予定するプログラムのことである。

線条的なプログラム理論は，スキナーの著作の中で完成した。もっとも単純な線条プログラムは，LL 教室用の資材で実現され，次の形式で表される。



線条的プログラムは、学生全員に同一の学習課程の実行を提供する。この種のプログラムでは、しばしば「分脈の通過証」の導入法、即ち、線条的プログラムの学習課題の実行が提示された構造の中で学生が通行証を書き上げるように導かれる。このようなものとして、ハンガリーの教師により作られた“ЛИМЕ”と“МЕСТОПРИ”がある。

“ЛИМЕ”は学習の意志伝達面での要望を満たすマイクロ状況を学習者に提供する。学習者は、ディスプレイに現れる画像に対応する代名詞や名詞を入力せねばならない。

Ты не встречался с..... (Юрой)?

君は (ユーラ) に出会わなかったのかい

Нет, я не видел..... (его).

いいや、ぼくは (彼を) みかけなかったよ

“МЕСТОПРИ”では学習者が学習を望む格の使用について格を選択することになっている。その一方、コンピュータはさまざまな文を生成し、それらをディスプレイ上に提示する。分脈に応じて必要な形容詞語尾とそれに対応する

人称代名詞を入力することが要求される。

Мы пригласили молод.....(ых) инженеров.

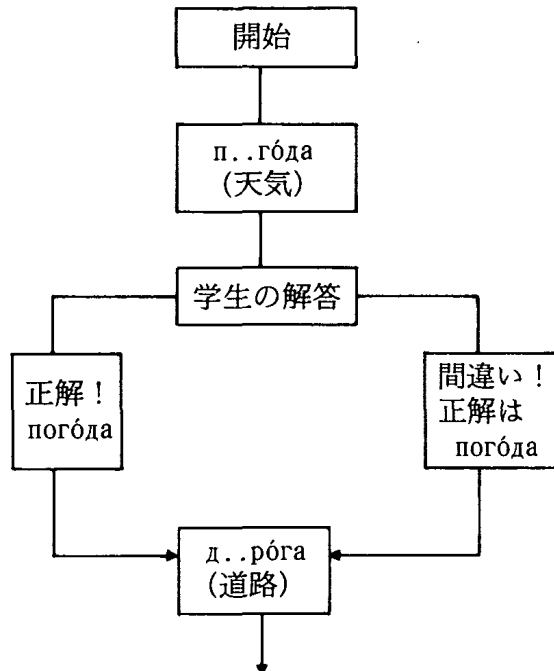
私達は若い技師を招待した

Мы пригласили.....(их).

私達は（彼らを）招待した

ハンガリー製のプログラムが明瞭に表された線条的性格を有しているにも拘わらず、そこには、やはり、学習者に一助を与えることが意図されている。

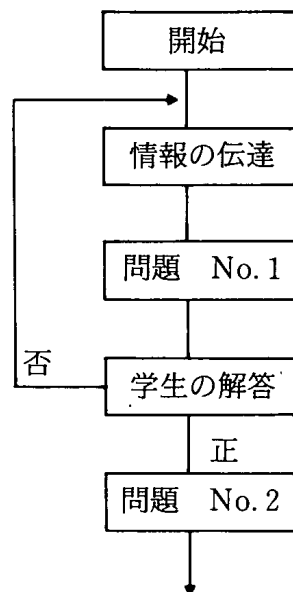
コンピュータ学習に際して線条的プログラムは、錬成用と試験用としてのみ使用されるのがその目的に叶っている。このようなプログラム例としては、非アクセント母音の正しい綴りを訓練する線条的プログラム「空いた文字を補いなさい」がある。



このプログラムのフロチャートから明らかなように、学生は正答を得た場合に次の課題に進むことができる。解答が間違っていた場合コンピュータは間違いのあったことの信号を出し、すぐさま正解を出す。然る後に学習者は次の課

題に進むのである。

教授法に関する文献には若干異なる線条的プログラムの理解も見られる。ここでは学習者が先行する問題の正しい解答の場合にのみ次の課題を受領するのだ。間違った解答の場合コンピュータは学習者を出発点の問題か、所与の問題に先行した情報の伝達に引き戻すのである。



情報伝達を2回目に読み込んだら（またはそれがなくとも）学生は再び最初の問題に引き返されて、自分の間違いを自力で説明すべく努力を行い、その後、新しい解答を入力する。

このようにしてプログラムに従った学生の解答は既に彼らの解答（正否）如何に左右されることになり、我々の見るところ、この種のプログラムは線条的というよりは、単純な分岐化プログラムに近い。実際、厳密に線条的なプログラムは、上記に述べたように学習者の解答に依存せず、出発点の素材に舞い戻ることなく、学習者全員を次のステップに向かわせるのだ。ここで、学習者は自分で自分の間違いを考えようと試み、正しい解答を出し、「肯定的」枝節、即ち次の問題に移る前にそれと同じ「否定的」枝節を何度か通過することが有り得る。この場合目的に到達する唯一の方法は、同じステップを何度か反復、通

過することである。それ故、この種のプログラムを循環的 (циклический) と呼ぶことができる。

線条プログラムは、外国人のコンピュータによるロシア語学習の際に既に指摘したように鍛錬と試験の局面においてのみ意義を持ち得る。なぜなら「低能」であり、コンピュータの可能性を最大限に利用することがないからだ。コンピュータは、周知のように、高度の技術手段であるから、その適用はより高度の課題解決に適している。然るに、線条プログラムはより単純な (プログラム学習と管理のような) 学習マシンの助けをかりて効果的に実現される。Ритм (リズム), Аккорд (和音), КИСИ, СУОК マシン。

N. Krauder の正鵠を得た指摘によると、「プログラムとは、より高度ではあるが、学習者にとっては許容される抽象度で書かれた素材に取り組むことを可能にしつつ、学習者の認識を拡大、深化させるようなもので、もっとも劣る者を見込んだ道筋に全員を追い込むものであってはならない」。

分岐的プログラムが、正にこのタイプのプログラムにあてはまる。

分岐プログラムは、学習者の基本的方向性 (活動) からのいくつかの分岐を持ち、そこでは入力される解答に対応する解説の予見が予定されており、学習者の先行するコマ (情報分量) への復帰、もしくは前進移動への奨励や許可は、逐次的またはコマ毎におこなわれる。

分岐プログラムでは学習情報の提示が、通常、学習者が犯す間違いに依存する。コンピュータに不正確な解答を入力すると、分岐プログラムでは、基本的な設問の本質の理解を助けるような補足的、もしくは誘導的な質問、解説、もしくは指示が予定されている。

分岐プログラムには2つのタイプがある。

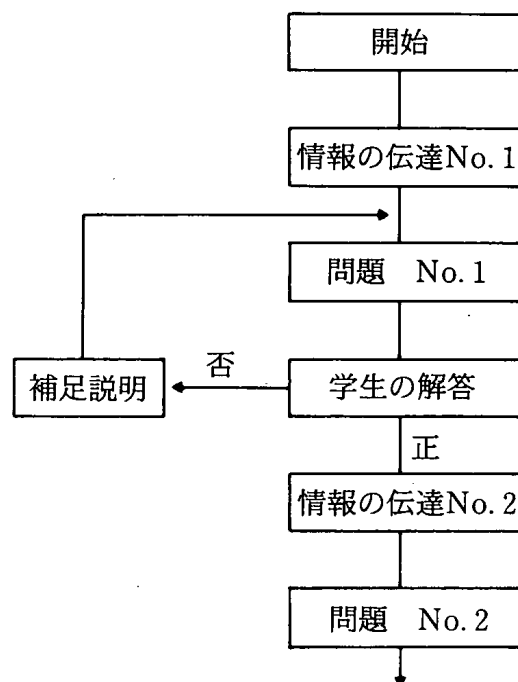
1. 内的に制御された
2. 外的に制御されたプログラム

内的に制御されたプログラムでは、ステップの順次性がプログラム作成者によって予め規定されている。外的に制御されたプログラムでは、続行するステッ

プログラムが学習者のこれまでの解答の統計に基づいて規定される。このタイプの分岐プログラムでは、次のステップを選択するためにコンピュータの統計手段が学習者の回答全体をまとめることによってプログラムを変更できるような計算をおこなっている。

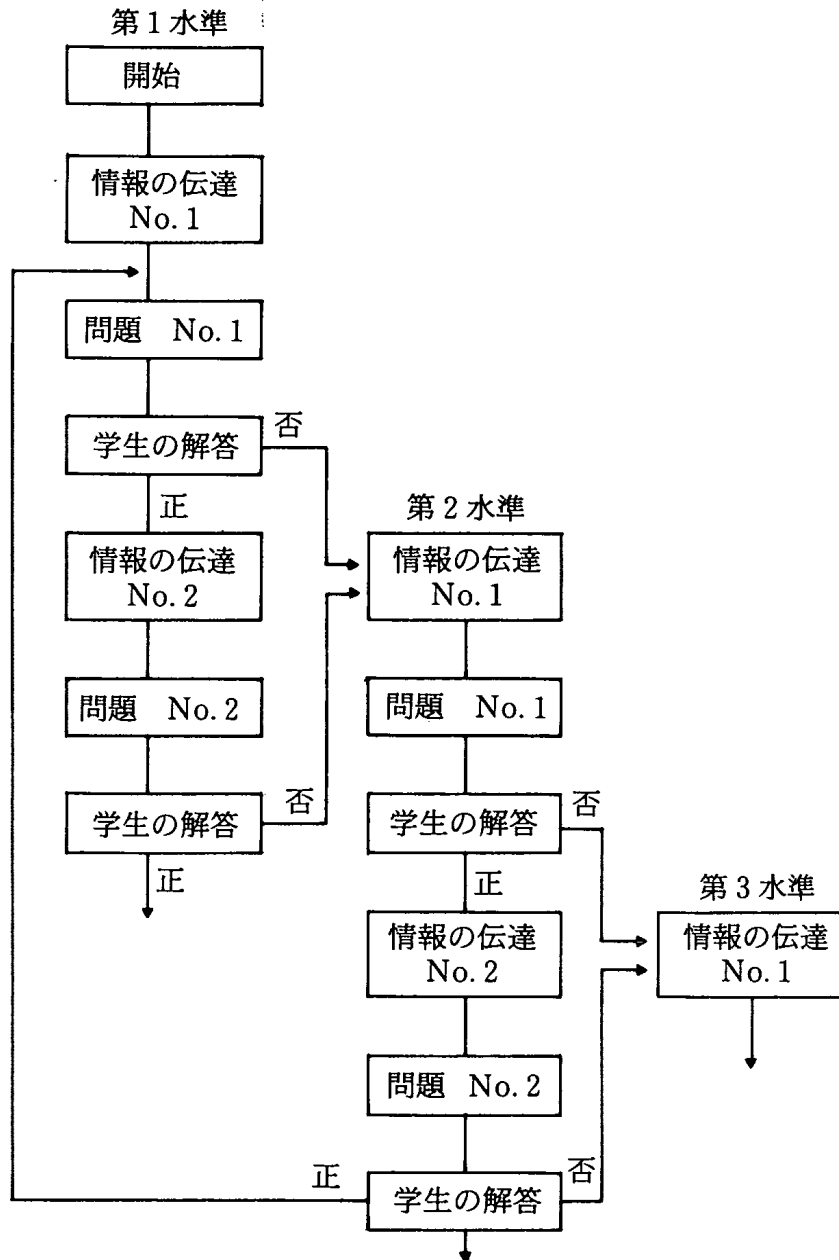
内的に制御されるプログラムには2種類ある。

- a) 一種類の記述水準と多くの補足ステップ(分量)をもつプログラム。枝分かれを伴わず、基本的水準の一定段階での質問に対して不正確な解答を出した場合、このプログラムは学生を補足ステップに振り分ける。補足的教材を学習した後、学生は振り出しの段階に戻ることができる。この教材の習得度に応じて、この手順は一度ならず反復される。枝分かれを伴わない、内的に制御されるプログラムのフロチャートを以下に示す。



- b) 枝分かれを含むプログラム(いくつかの記述水準を持つ)。難易度の選択は、学習者の知識に依存する。さらに、プログラムの実行中に学習者は自分の知識に応じて難度から容易な水準に移行、または逆の動作をすること

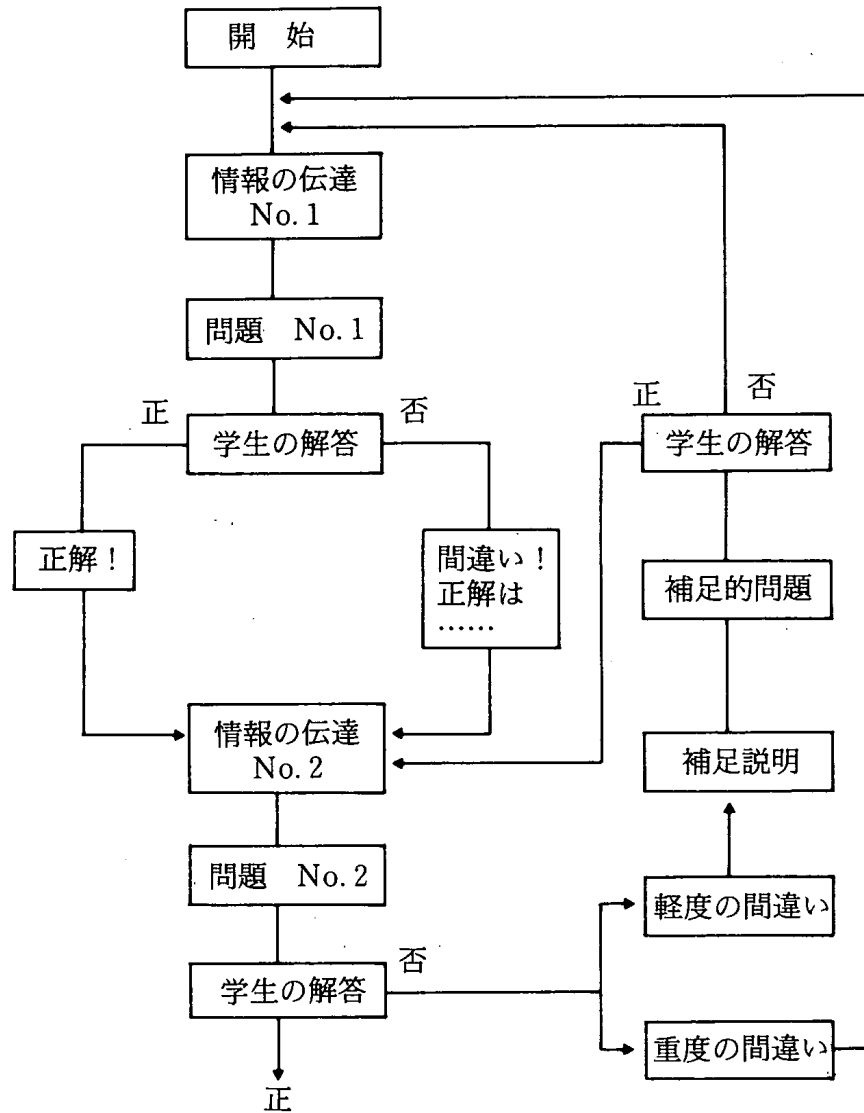
が可能である。



かくして「出来の良い」学生は、分岐プログラムの著作者によって規定され、ロシア語事象の習得に必要とされる最小限の教材を勉強する「出来の悪い」学生に比べて、より高度の教材を勉強する可能性をもっている。

分岐プログラムは、プログラム作成者が設定する具体的な課題に応じた、極めて多彩な構造をもっている。

混合プログラムは、線条的と分岐的な部分を内包しており、まさにそのことによって学習の柔軟性を保証している。学習体験によりこの種のプログラムでは、学習者の個人的能力に合わせることができる。



学習者の特性に順応するプログラム能力に応じて

1. 順応性
2. 一部順応性
3. 非順応性

プログラムに分類される。

順応的プログラムは、学習の外的、内的条件（先行する教材の程度や速度など）により、教材の記述方法を交換することのできるプログラムである。このようなプログラムは、通常、同一機材でも記述の方法や適合プログラムの積極的選択の制御手段の異なる、いくつかの線条、分岐プログラムから構成される。他言すれば、順応プログラムでは、出された質問に対する学習者の解答の充足度や正確さに応じてコンピュータが選び出す質問、課題、解説、推奨がいくつか用意されている。

かくして、順応プログラムはもっとも完全に学習者の知識、能力、習慣付けの程度を考慮していることになる。順応的プログラムの使用は、外国語としてのロシア語の全学習課程で目的に叶っている。

一部順応的プログラムは、全ての領域で適合する性格を有しているわけではなく、プログラム作成者の意見によれば、学習者にとってもっとも難しいとされる一定段階にふさわしい。

非順応的プログラムは、同一箇所の教材学習を学習者全員に提供する。この種のものとして上記に見てきた線条的プログラム全てがあてはまる。

構成要素数と配列順序によって、次のタイプに区分される。

1. 単一要素の構造をもつ
2. 二要素の構造をもつ
3. 三要素の構造をもつ
4. 多要素の構造をもつプログラム

単一要素の構造をもつプログラムは、質問と課題、与えられる解答が単一の全体をなしてる。たとえば、造格の練習を指向する選択解答式のプログラムがある。

Я был в кино с..... подруги (生格)

подруге (与格)

подругой (造格)

подругу (対格)

подруга (主格)

同種のプログラムは、任意のタイプのコンピュータで容易に実現されるが、この種のコンピュータプログラムの作成と利用は、ほとんど目的に適さない。なぜならこれらはより単純な教育機器のためのプログラムとして作られたからだ。

1. Аккорд

2. Ритм

3. Альтернатор

コンピュータ用プログラムの作成には、常に「外国語学習におけるマイクロコンピュータの利用」が他の方法や他の手段によっては解決の困難、もしくは不可能な目的や課題を実現にできるような場合にのみ推し進められることを覚えておかねばならぬ。その他、単一要素プログラムは、教授法上の多くの欠点をもっている。即ち、

1. 単独の解答形式を保証しない

2. まちがった解答に学習者の注意を引きつける

3. コメントと説明をおこなうことが不可能である

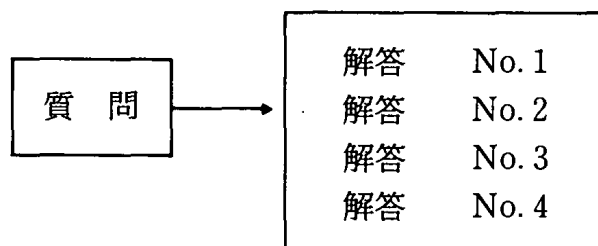
4. プログラム作成者の作業を複雑にする (提示される解答に認められる些細な違いにも考慮する点で)

二要素構造のプログラムでは、課題と解答の提示はすでに分離されて、継起的に発生する。

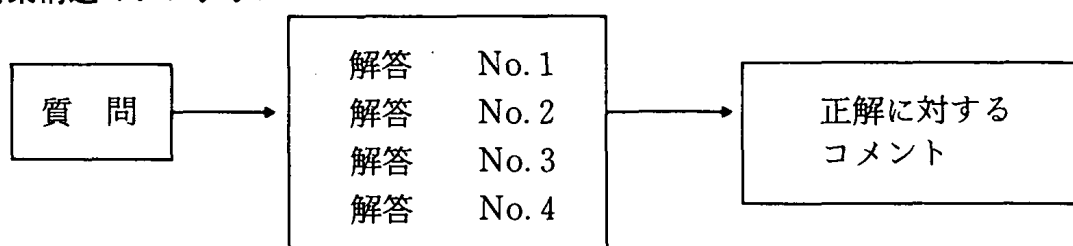
3要素構造のプログラムは、前者の発展タイプとみなすことができ、正解のコメントが付加されるにすぎない。

とはいえ、質問や課題、解答をバラして提示するにも拘らず、二ないし三要

## 二要素構造のプログラム



## 三要素構造のプログラム



素構造のプログラムでも単一要素プログラムがもつ欠点が除去されているわけではない。

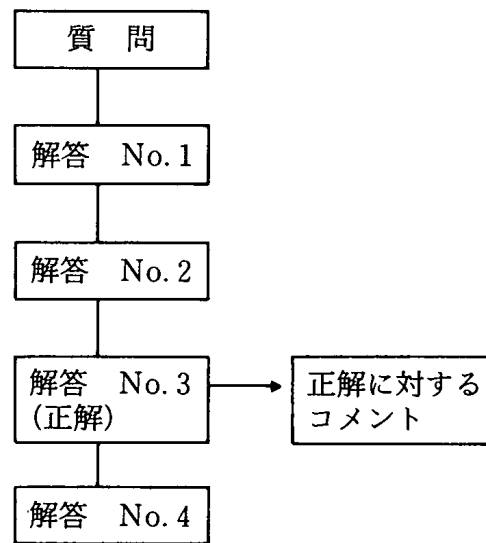
多要素構造をもつプログラムは、全段階の分離を特徴とする。これには出される解答の分離も含まれる。この種のプログラムではまずディスプレイ上を泳ぐように解答が次々と明滅する。「遊泳」の速さ(速, 中, 遅)は, ディスプレイ上を「泳ぐ」解答の中から正解を選択することになる学習者自身によって決定される。解答を選択した後に, ディスプレイには正解のコメントが映し出される。

多要素構造のプログラム例として, ハンガリーの著者グループによるプログラムがある。

“ОТРИЦАНИЕ” (否定)

“ПОЙМАЙ СЛОВО” (単語を捕まえよ)

“ПОЙМАЙ СЛОВО” で学習するとき, 学習者はディスプレイ上泳ぐ単語を「捕まえながら」文を正しく作っていかねばならない。各々の試みを行った後で,



コンピュータは現実的区分に基づいた可能な文全てを学習者に提示する。ハンガリープログラムの疑いもなく素晴らしい点は、それが遊びの様式の中で解答されるというところにある(漁師は学習者、魚は単語)。とはいっても、同時に、共通する欠点、すなわち、文を組立てるのに既に出来上がっている形式を学習者に提示していること、これがプログラムの上では選択の(往々にして偶然の)要素が優勢になり、文を組み立てる際自力で文法形を形成するに至らないことにつながる点である。

我々の見るところ、より効果的なのが、多要素構造のプログラムで、その第1段階では学習者が文の組立に必要な単語の最初の綴りだけを「捕らえ」、その後「魚」をうまく捕らえたら、すなわち、必要な単語を全て選び抜いたら、第2段階に移行する。これこそが創造的性格を備えているのだ。この段階で学習者は、語頭の綴り適切に切り換えながら、選びだした単語から自力で文を形成していく。

教授法上の目的から何種類かのコンピュータプログラムに分類することができる。

1. 文法的、即ち、ロシア語文法の習得をめざす(性、数、格、人称、動詞

の体、語尾変化など)

2. 語彙的(同根語, 同意語, 反意語, 同音異義語などを捜せ). 語形成に関するプログラムも, これに加えることができる.
3. 言語文化的, 即ち言語文化的側面とそれに基づく情報の研究に寄与するプログラム.
4. 読み方や書き方のような発話行為の学習をめざすプログラム(正書法, 句読法プログラムなどがある)
5. コミュニケーション. 音声学上(コンピュータがタープレコーダや発声器と一体に作動する場合に限り有効)

現在の外国語教授法の重要原則の一つに, 総合性(комплексность)がある限り, 上述したプログラムはしばしば複合的な使われ方をする.

プログラムの独立度によって

1. 自律的(独立的)
2. 局所的(連合的, 依存的)プログラムに下位区分される.

自立的プログラムとは, 特定の教科書や参考書に結びつかないプログラムで, それだけに, 外国語としての様々なロシア語教科書の学習に使用することができる. このようなプログラムは, 何よりも, 学生の独習を目的としており, 訓練的な性格を持つ. 何らかの特徴(格語尾の用法練習など)を基軸に自律的プログラムを特定のシリーズや複合体に統合することが可能である. しかしながら, これは, このような複合体の任意のプログラムが他のプログラムとは無関係に利用可能であることを前提にする.

局所的プログラムは, 言語資料の統一バンクを基に作成するのに適している. この種のプログラムの利用は, 外国人学習者には習熟や能力の形成がまだ始まったばかりの初期の学習段階で肯定的な効果をもたらす. それ故に, この段階でのコンピュータプログラムは, 基本教材と自然な形で結びついているようにすることが特に重要である.

開発の終了した任意のコンピュータプログラムは, プログラムのバグの有無

を発見する目的で試行手続きを踏まねばならない。

試行の複雑さに応じて、以下のプログラムに分類可能である。

1. 単純
2. 中位
3. 複雑
4. 高度に複雑
5. 超複雑プログラム

単純プログラムは、1,000 ステップ以下のソースリストをもち、通常、1 人のプログラマーによって最長 6 カ月間で開発される。この種のプログラムは、通常他のプログラムやシステムに連動しない。

中位の複雑さをもつプログラムは、1 万ステップ以下のソースリストを含む、その開発には通常 1 から 5 人のプログラマーが参加する。開発期間は 2 年を超えることがない。このプログラムは他のプログラムとの関連が多少程度あるか、もしくは概してない。この種のプログラムの試行作業には 6 から 12 カ月間とられる。

複雑プログラムは十萬ステップ以下のソースリストを含み、5 から 20 人のプログラマーチームが 2 から 3 年間で開発する。この種のプログラムは、通常いくつかのサブシステムから構成され、しばしば他のプログラムと共同作業をおこなう。このようなプログラムの複雑さ故に、一つのバグ位置の発見と修正には、2, 3 日のことも、数週間（時には数カ月）かかることも有り得る。

高度に複雑なプログラムは、百万ステップ以下のソースリストを内包し、百から千人のプログラマー集団により数年間をかけて開発される。この種のプログラムは、通常複雑な相互関係を伴ういくつかの基本サブシステムから構成される。持続的作業を遂行させるために、初期段階の作業集団に加わっていなかった専門家の誘い込みが求められる。

超複雑プログラムは、百万から千万の命令を含む。開発には千人のプログラマーを超える大集団が要求される。この種のプログラムの開発期間は、十年に及ぶ。このようなプログラムは、とてつもなく大きな期待を有するものであら

ねばならない。なぜなら危機的過程(航空輸送の運航管理, 防空システムなど)とまず例外なく結びついているからである。

試行の複雑さという観点からのプログラム区分に照らしてみれば, 現行の外国語としてのロシア語のコンピュータプログラムは, 通常単純プログラムのそれであり, 中位の複雑さをもつプログラム開発の点ではささやかな歩みを踏み出したにすぎない。

### コンピュータプログラム作成の教授法

コンピュータプログラム作成に着手しようとする教師なら外国語としてのロシア語教授のみならず, 本書が目的としている具体的なコンピュータ技術の可能性からも出発せねばならない。

通常コンピュータプログラムに含まれるのは,

- 1 教材
- 2 学習者の状態の診断手段
- 3 学習者の解答と動作の分析手段
- 4 学習者の知識, 能力, 習熟の評価手段
- 5 学習者の学習に関する統計資料の収集と分析の手段
- 6 学習者の行動の管理手段

コンピュータや自動化された学習システムのタイプに応じてコンピュータプログラムのいくつかの要素は, コンピュータの技術的蓄積の手段によってプログラム作成段階では自動的に構成に組み込まれる。

コンピュータプログラムの作成は, 通常 3 つの基本的段階に分かれる。

- 1 教材の教授法上の研究
- 2 学習者の行動を管理する過程のアルゴリズム化

### 3 学習・教授法上の素材のプログラミング

最初の 2 段階はロシア語教師によって実行され、3 つ目のみがプログラマーの手にかかることは極めて明白である。無論、高級言語の一つで純粹のプログラミング作業を全ての教師に教える必要はない。とはいえ、プログラミング能力をもつ教師なら、それをもたない同僚に比べはるかに少ない時間的ロスですませられるのだ。

プログラム著者達は、それを編集する際に、実物のコンピュータや実際のプログラミング言語の限界を明確に把握して、プログラムには実物のコンピュータで実現されるだけの教材や順次性を盛り込むようにせねばならない。

それではコンピュータプログラム生成の諸段階を検討することにしよう。

- a) コンピュータの応用目的
- b) ロシア語課程におけるコンピュータの位置付け
- c) 学習者の開始時の知識程度
- d) 学習者のコンピュータ利用学習の結果えられる終了時の知識程度
- e) 設定された目標の達成基準

この段階で教師が心得ておかねばならぬことは

- a) コンピュータプログラムのための教材を選定すること
- b) それを各部や断片に分類すること
- c) これの提示順序を決定すること

コンピュータプログラム作成の第一（準備）段階を終えたなら、第二（基本）段階、つまり学習者の行動を管理するような過程のアルゴリズムの段階に取り掛かる。この場合、アルゴリズムとは、コンピュータによる学習者の将来的作業のシナリオと理解できよう。

コンピュータプログラムのアルゴリズムに含まれるのは、

- 1. (一定の順序で部分や断片に選定、分類された) 教材
- 2. 学習者の行動の予測
- 3. 学習者の解答や行動によってディスプレイに出力されるプログラムの

反応（テキスト）

4. 学習者の行動の評価
5. コンピュータに登録蓄積される統計資料の一覧表
6. 一つのプログラム断片から次のへの移行条件と順序

任意のコンピュータプログラムの最小単位は、コマ(кадр)である。コマとは、論理的に完成した一区切りの情報、質問や課題、問い合わせ、解説、ヒント、評価などを含むコンピュータプログラムの一要素と理解すべきである。コマの他にも квант, доза, порция 等いくつかの名称がある。

しばしばコンピュータプログラムでは次のタイプのコマが用いられる。

情報
管理（試験）
照会
解説
指導（ヘルプ）
評価
操作（補助）

情報コマは、通常何らかの学習情報を内包しており、導入、指導、概括、試験に下位分類される。

管理コマは、各種の設問や課題を含む。

照会コマの使命は、学習者の要望に応じてディスプレイ上に何らかの問い合わせを出力することである。

解説コマは、ヒント、解説、各種コメントを内包する。

指導コマは、個々のコンピュータ教材をいかに学習するかについての情報を学習者に与える。

評価コマは、個々の応答やコメントを含み、学習者の何らかの行動、または

学習全体が評価される。

操作コマは、受け取った情報の改変を指導し、学習者に提示された課題の順番を決定する。

現実のコンピュータプログラムでは全てのコマが、通常有機的に結合している。その他、30 分以上のコンピュータプログラムには、いわゆる弛緩プログラム（冗談めいた情報、遊戯的プログラム、アニメ的な画像、音楽など）を必ず入れなければならない。

コンピュータプログラムは各々専用の表題に始まり、学習者はこれからのプログラムの用途や著作者名、制作地の情報を得るのである。

次の段階のプログラムの便宜のために教材は次表のように配置するのが適切である。

コマ番号	内 容	正解と移行 方向	誤問に対する 表記と移動方 向	援助評価	特記事項
------	-----	-------------	-----------------------	------	------

それでは外国語としてのロシア語のコンピュータプログラムの具体例を見てみよう。基礎には М. И. Калинин 名称 ЛПИ (レニングラード教育大) の準備学部で作成され、言語文化的教育目的と局所的、すなわち、ソ連邦の大学の準備学部で使用されている「Старт-1」の特定のレッスンを目的とし、既に教育課程では公式認定を受けているプログラムを据えることにしよう。

このプログラムの目的は、外国人学生に БАМ, Сибирь, ССО, третий семестр のような国情研究用語彙を教えることである。プログラムによる学習をおこなえば、学生はこれらの単語を認知、理解、正しく使用できるはずである。

教授法では通常語彙に関して次のタイプの学習に分類している。

1. 語彙との最初の遭遇、導入もしくは提示
2. 意味化 (семантизация)
3. 導入語の理解度の測定

4. 強化

5. 活性化 (активизация)

6. 総括試験

このプログラム作成に際し、我々は一般に受け入れられている図式から意識的に少し逸脱した。コンピュータプログラムは学生に何らかのことを教えるのみならず、学生がどの程度予習して来たかを測定することもできるのである。学習者が既に動詞の過去時制と名詞の前置格（この文法事項 Старт-1 の先行レッスンで学習済み）を知っているので、学習者がどれほど既習教材をマスターしたかを測定するのが第1要件である。かくして、コンピュータプログラムは新語の提示から始まるのではなく、学習者がコンピュータによる一連の予知的質問に対する解答からなのである。コンピュータは次の質問に答えるよう求める。

「あなたの名前は」

「現在何という都市に住んでいますか」

「あなたはホテルもしくは寮住まいですか」

「あなたは仏語（西語，英語）が話せますか」

「動詞 работать の過去形を書いてみてください」

質問は学生の解答の後次々と機械からだされる。従って質問が全くの偶然的なものに思われようとも、各々の質問完全に一定の目的をもっているのだ。例えば最初の質問に対して学生が自分の名前をコンピュータに入力したら、その結果として次からは（学生の完全な満足が得られるのであるが）コンピュータは学生に対し名前で問いかけるのだ。こうしてプログラムは特定人物の利用を対象としないプログラムをやめて、事の初めから学習者一人一人のためのプログラムに変貌し、生きた交流という幻想を生み出す。

2 番目、3 番目の質問は、学習者の名詞の前置格形の知識と理解度を測定する目的で提示されている。4 番目の質問によって必要とあらばコンピュータは

何語で学生に説明やヒントを与えるべきかが明らかになる。(このプログラムは 3 カ国語の使用を前提にしている)。5 番目の質問は動詞の過去形の知識を点検する。

全ての解答がコンピュータの記憶にとどめられ、コンピュータはそれを基にコンピュータ基本プログラムの学習に対する学生の準備程度の結論を下すのである。誤答数が多ければ、コンピュータは必要な文法教材を復習するよう学生に求め(この場合専用の復習プログラムが起動する)、誤答がない、もしくは少ない場合には学習者は基本プログラムの学習に取り掛かることができる。

基本プログラムは、4 つのレベルの民族文化的語彙要素を含むマイクロテキスト (初期の教材提示) によって開始される。

Летом Виктор работал в Сибири на БАМе. Мы тоже работали в ССО.  
Это был наш третий семестр.  
夏ビクトルはシベリアのバム鉄道で作業した。我々も ССО で働いた。  
それは我々にとっての第 3 学期であった。

学生にマイクロテキストを提示したら、コンピュータは質問を与える。「あなたはこのテキストでどんな新語に気がつきましたか」

学生がキーボードから新しい国情研究用語彙を入力したら (その他の単語については既習單元か復習用サブプログラムで知っているのもので)、コンピュータはロシア語と選択言語の双方で平行的に説明を行いながら、新語の意味化を促す。

注) 文化的背景をもつ単語のもっとも適切な提示方法は、単語の意味解釈と意味背景の説明であろうことは経験的に証明されている。

意味化の後、コンピュータは導入された国情研究用新語の理解度測定に移行する。学生には一連の、今し方読み終えた教材に関する質問が出される。

「ССО とは何か。」「БАМ とは何か」「Сибирь はどこに所在するか」

学習者の行動管理の手順化に際して、重要な要素になるのが学習者の動作に

対する予測である。教師は学生の可能な限りの解答を全て予想し、各々の解答に対応する反応テキスト、すなわち、正解もしくは誤答のときディスプレイ上に現れるテキストを用意しておかねばならない。

「COO とは何か」の問いに対して、学生は「学生建設隊」と答えるかもしれない。これが正解であるなら、コンピュータは対応する反応文を掲げなければならない。「あなたは問題に正しく解答しました」。学生が誤答をおこなえば（「ソビエト式建設隊」）、反応文は、当然異なる。「残念ながら、間違いです。ソビエト式という単語は、多分『ソビエト連邦』誌や『ソビエトスポーツ』紙の名称からもってきたのかもしれませんが、我々がここで問題にしているのは学生の働く建設作業隊のことです。もう一度この質問に答えて下さい」

その他の誤答を検討しよう。「ソビエト社会主義隊」  
ここでの反応文はさらに異なる。「たいへん残念ですが、あなたの解答には間違いがあります。なぜなら COO と CCCP（ソ連）を同列に置いているからです。よく考えてから正しい答を出しなさい」

学生の解答を全て予知することなどは不可能であるから、コンピュータプログラムを作成する場合の教師は常に予知できなかった解答に対する反応文も準備しておかねばならない。「失礼だが、このような解答は我々のプログラムにはありません。もう少し考えてから、別の解答をだしてください。\*」

\* (訳者注)

この種のコメント文は 1991 年現在既に「廃れて」いる。ハードソフトウェア両面からの進歩が著しく、また、ユーザもコンピュータの「従属」に甘んじる時代ではなくなった。もう少し賢い反応、融通の効く対応を期待する時代になった。

あれこれの質問にたいする解答の中でもっとも典型的な間違いは、プログラムによる学習の最中に発生する。学習終了後に学習者全員の学習記録をみて、教師は学生の予知できなかった解答を必ず分析し、間違った答に分類されるよ

うな予知不能の解答に対するテキストを新しく書き込むなどしてコンピュータプログラムを再編集しておかねばならない。

コンピュータプログラムを作成する際の大変重要な問題として、プログラム素材の収集過程での現有収集物（これまでのソフトウェア財産）の組織化がある。学習マシンのための学習プログラム編成法や Н. А. Омельченко В. Я. Ляудис が収集物の組織化法の解明のために実現させた方法論、我々のプログラム化参考書の編成法などの文献を分析してみると、矯正の影響は、基本的に学習者の進歩如何によって左右されることがわかる。ほぼ全ての学習のプログラムで遅滞の無い矯正情報の提示が、例えば正解の手本とか、補足的説明という形で実現されている。

А. Н. Леонтьев や Я. П. Пономарев のような心理学者の研究が示しているように、収集物の組織化の際重要な役割を担うのが、主観的要素—ヒントを提示する間合いである。「具体的な形のヒントも、課題とヒントが時間的に接近していても、ヒントを与えて行動の筋道を具体的に指示しても、行動の遂行条件を具体的に指示しても、これは決定的状況でない。決定的なのは、ヒントの提示時期なのだ。つまり、もしヒントが課題に先行すれば、これの効果は薄れてしまう。一方、ヒントが課題の後付けになれば、学習者に課題を繰り返して喚起する場合かなり著しい肯定的効果を与えるであろう。」

ここで検討されている国情研究用語彙のプログラムは、一つの問題につき 3 回までの誤答を許容する。然るのちに学生に対しあれこれの説明を再読し、それから正しく解答するよう求める。誤答表示のシグナルからコンピュータによる矯正情報の提示までの時間は、学生自身の搜索動作によって埋められる。

とはいえ、準備学部での外国人学生のコンピュータディスプレイ上の学習教室で授業運営してみると、一つの問題に対して解答を求める多くの試みが期待した効果をもたらさないのだ。2 回目の誤答を出した学生は長いこと考え込んでしまい、事実上、暗礁に乗り上げている。外国語としてのロシア語の初期の学習プログラムを作成する場合、一つの設問に対しては二回の解答チャンスに制限するのがよろしい。学生が正確に答えられるまでプログラムが次の設問に

進まないようにして、「設問—コメント・ヒント—設問」という閉塞的循環に学生を手懐ける。正答に限り、この閉じたリングから学生を解き放つのである。

上記のマイクロテキストに関する設問全てに正答を得たら、コンピュータはもっと手の込んだ強化テキストを学習者に提示する。このテキストで学生は必ず今し方マスターした言語文化的試元に遭遇することになる。周知のように、「読書は学習手段でもある」のだ。様々な組み合わせで言語素材を何度も反復することは、それをもっとよく覚えること、従って、他の言語活動の場にあってもそれをもっと広範に使用する可能性を広げてくれることになる。

以前検討したマイクロテキストと同様、新しいテキストによる学習は「断片指向システム」(Фрагментарно-ориентированная система)の原則に即して組み立てられる。このシステムの著しい特徴は、一コマの移動が、先行するコマの試験問題に対する解答の良否に左右されることだ。

プログラムに多様性を持たせるためにテキストを対話に切り替えることができる。マイクロテキストによる学習が終了すれば、ディスプレイ上に人間の画像と発話が現れる。

1. こんにちは！ 私の名はアレキメデスといいます。あなたは。

(学生の返答)

2. 私は外国人です。ぼくはアルジェリアから来ました。あなたは。

(学生の返答)

3. 私は読書がたいへん好きです。ところであなたは読書が好きですか。

(学生の返答)

4. ぼくは昨日「ソビエト連邦」誌で興味深い記事を読んだけど、BAM と CCO の2語が判らなかった。あなたならこの単語を知っているかもしれない。これは一体何なのですか。

(学生の返答)

5. ありがとう。今度はぼくも BAM がバイカルアムール鉄道で、CCO が学生建設隊ということがわかった。夏にはきっと BAM の CCO で作業しに出かける

ことにしよう。ところであなたは CCO で作業するつもりですか。

(学生の返答)

コンピュータ「アルキメデス」と学習者のこの対話は、極めて多様な性格をもち得る。この対話の最中アルキメデスは学生に CCO と BAM の追加情報を知らせる。第 4 問にたいする否定的回答(「知らない」「書けない」「忘れた」「違う」「覚えていない」など)の場合、アルキメデスは自分とともに書物内の新語の意味を調べるよう提案する(ディスプレイ上には 2 カ国語にて説明文が出力される)。

先行する課題をすばやく、且つ正しく解いた学生にコンピュータは以下のような一連の練習問題を解くよう提示する。

1 次の語から文を作りなさい。

- A) рабтать, БАМ, летом, студенты, на
- B) строить, ССО, школы, дома, больницы
- B) третий семестр, любить, студенты

2 次の名詞と結合する形容詞を選択せよ。

- A) теплый (暖かい)
- студенческий (学生の)
- отряд (部隊)
- холодный (寒い)
- строительный (建設用の)
- B) длинная (長い)
- весёлая (陽気な)
- новая (新しい)
- МАГИСТРАЛЬ (幹線)
- Байкало-Амурская (バイカルアムールの)
- больная (病気の)

### 3. CCO について語るために必要な単語と語結合を選択せよ

киоск (キオスク), школы (がっこう), каникулы (ちょうききゅうか), клубы (かいかん), яблоко (りんご), тетрадь (ノート), третий семестр (だい3 がつき), строительный (けんせつようの), работать (はたらく), цирк (サーカス), строить (けんせつする), студенты (がくせいの), отряд (ぶたい), Сибирь (シベリア), студенческий (がくせいの), в (きかんちゅうに), отдыхать (きゅうそくをとる)

「学力のある」学生に対してコンピュータは選択した単語から CCO の語を書き上げるよう求める。

学生のプログラム学習が効果的であるために教師は上記タイプの練習問題の施行に伴う可能な限りの全ての正解を予測し、反応テキストを作成していなければならない。第1のタイプの練習問題をおこなう際、学生の答えそうな例として

「夏、学生たちは БАМ で働きます (した)」

「学生たちが БАМ で働いたのは、夏のことでした」

「БАМ で学生たちが働いたのは、夏のことでした」

もし文の現実的区分に注意していなければ、これらの解答は全て正解になる。それ故、コンピュータの記憶にはこの場合一つの標準規格ではなく、正解をセットで保存しておかねばならない。もっと広範に可能な誤答を想定しておかねばならない。複合的な教材展示をする場合の教授方上の原則に従い、文法的間違いをも予測せねばならない。この場合は格の用法と動詞の変化。学生が文法的間違いを犯したら、コンピュータは反応文を学生に発する。その中で誤りの性質を示し、それを訂正するよう求める。一例に学生たちが時折次のような文を作る。

「学生たちは夏の間 БАМ で働く こと」

コンピュータは遅滞なく誤りに反応せねばならない。

「間違っています。動詞 *работать* は、現在または過去時制にしなければなりませんのに、ここでは不定形になっています。もう一度教えてください」

2 回目も誤答の場合コンピュータは学生に文法の手引きを見せる、つまりディスプレイ上に種類の動詞の変化表や動詞の過去形作成法が出現する。このようにして、必要な助言をもらった上で学生は自力で正しい答を導き出すことができる。

仮に格の用法に間違いを生じた時には、

「学生たちは、バムが夏の間に働いている」

当然のことながら、ディスプレイには他の反応文が出現する。

「間違いです。BAM は主格になっていますが、前置格でなければなりません。他の解答を出してください」

2 度目の解答があれば、学生は文法の手引きに向かわされる。ここでは名詞の格変化の部分である。然る後に問題に対して正解に解答したり、課題を遂行することができるようになる。

プログラム学習の終了についても各々の学生にとってマチマチである。全ての課題を立派に迅速にこなした学生にはコンピュータの褒め言葉があり、感謝の意が表明される。

「全てのプログラムを非常に立派にこなしました。ありがとう！ さようなら！」

学習中にしばしば文法の手引きを参照し、それ故に、課題を全てやりこなすに至らなかった学生にはコンピュータがもう一度「遊びに」来るよう求める。

学生がプログラムで学習している間コンピュータは杓子定規に学生の解答回数、その際に犯した誤りの回数を記憶に蓄積し、秒単位の正確さをもって、学生が一定の解答やプログラム全体に要した時間を計測する。学生がプログラム全部を行うに至らなかった場合、コンピュータはこなした課題数をカウントする。その後コンピュータは得られた結果と教師によって予めプログラムに組み込まれていた評価基準の比較をおこなう。

もっとも単純な学習結果の評価法は、正確に遂行された課題の数によるもので、例えば、プログラム中に合計 12 題あれば、11～12 題の正解に対して機械は「優」の評価を下す。もし 9～10 題の正解なら、評価は「良」に下がり、7～8 題なら「可」、6 題以下なら「不可」となる。

しかしながら、このような評価システムでは、同一問題に対する解答試行の回数も、誤りの性格も、個々の問題や課題に要した時間も考慮されない。

著者の中には、学習評価もパラメータとして上達の相乗効果度を提起している。

$$P = N^1 / N$$

ここで  $N^1$  は、初答で誤りなく解答した回数

$N$  は、試行動作の総数

他の論文では、活動の評価にいわゆる習得パラメータを用いる。

$$t = t^2 / t^1$$

ここで  $t^2$  は、プログラム実行に教師が費やした時間

$t^1$  は、学生が費やした時間

我々にとってもっとも客観的に思われる評価システムは、採点数によるものだ。仮に初回の解答から正解に解された課題一つにつき 10 点を加算することとすれば、12 題の正答を得た時の総点は 120 点となる。誤答は各々、総点から 1～5 点の範囲で差し引かれる（5 点とは、このプログラムの学習目標の到達に係わる誤りに対して、他のタイプの誤りに対する「罰則点」は教師の裁量による）。

このようなシステムは解答時間も考慮される。例えば学生があれこれの問題や課題に 5 分間以上考え込んでいるようなら、以後 1 分間毎に罰則の 1 点を差し引く。

正答によって得られた合計点数をコンピュータが有りうべき総得点（満点）に振り分け、いわゆる習得の相乗効果度を導き出す。

$$K \text{ (習得度)} = K / K^1 \text{ (} K \text{ 割る } K^1 \text{)}$$

ここで  $K$  は、正答の合計得点

K<sup>1</sup> は、満点

コンピュータには、後、結果と記憶されている評価スケールの照合が残されているだけだ。

$0.95 \leq K$	習得度 $\leq 1$	優
$0.75 \leq$	$< 0.95$	良
$0.65 \leq$	$< 0.75$	可
	$< 0.65$	不可

これら全ての操作をコンピュータはものの数秒間におこない、その結果として学生の評価が画面に出力される。同時にプリンタが学生各自の学習記録を教師に出力する。

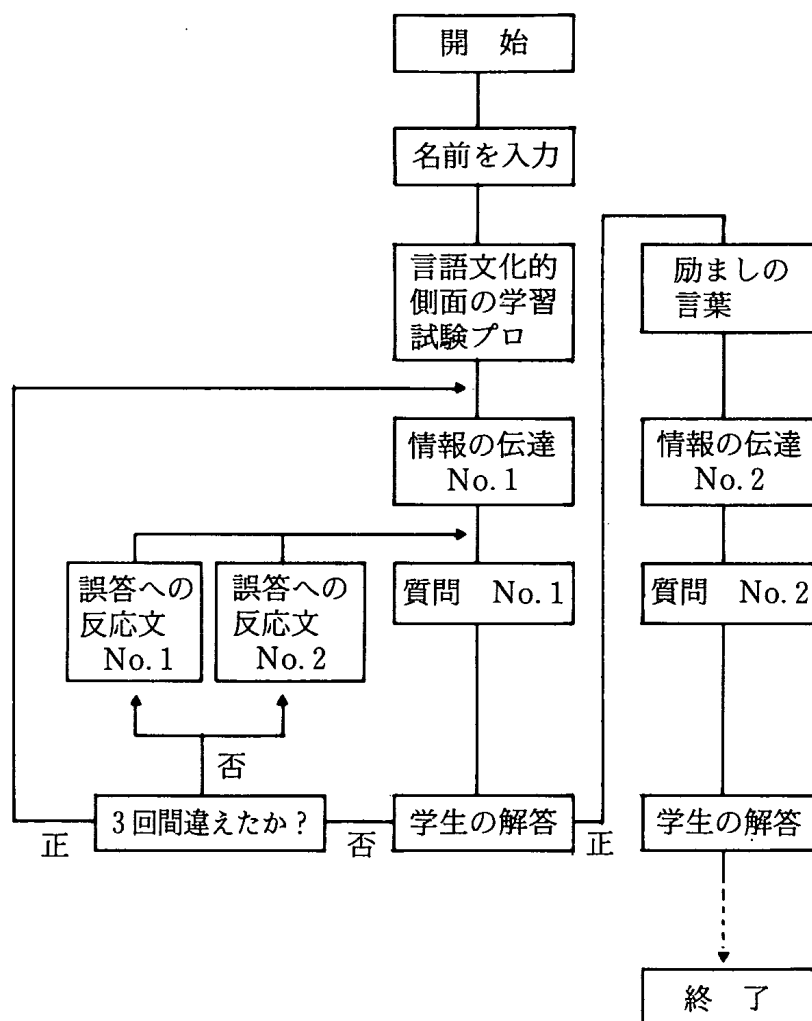
質問 番号	誤答	時間	試行 回数	誤答の 合計	評価
----------	----	----	----------	-----------	----

このようにログは教師が知識形成の程度と質を、また、教材の習得によって全体もしくは個別のレベルの程度を判断するのに役立ち、学習者の知識を深化、拡充すべきコースウェアの単元やテーマを明らかにする可能性を与えてくれる。つまり、学習の結果は将来の学習課題の制御に用いられる。

コンピュータプログラムを作成するときの教授法上の基本原則が理解できれば、今度は難なくこれまで検討してきたプログラムの総括ブロック図がお分かりであろう。

勿論真面目なコンピュータプログラムの作成は、非常に長期の、そして労力を要するものだ。約1時間の作動時間を要するコンピュータプログラムの作成には、その複雑度にもよるが、50 から 500 労働時間が必要とされる。

とはいえ、経験上分かるのであるが、もっとも労力のいる工程は、プログラムのシナリオ作成ではなく、またプログラミングそのもの（高級言語の一つでコーディング）でもなく、唯プログラムの試行とバグ取りなのである。プログラムの試行とバグ取りには、平均して、プログラム作成全工程の 1/3 から 1/



2 の時間が費やされる。

どのコンピュータプログラムでも学習パッケージ（自動化された学習コースウェア）にされる前に、全面的な試行とバグ取りがおこなわれ、そのことがあらゆる関係で高質、且つ、モノになるプログラムの創造を可能にしている。従って、「調整済みのプログラムとは、それが動作不能の状態になる条件が発見されなくなったプログラム」といわれるのも偶然のことではない。

プログラム開発に要する、従って、バグ取りに要する時間は、もしプログラム開発者が既に存在する応用プログラムを最大限に利用するなら、著しく短縮されるであろう。この種のプログラム例として、バグ、入出力、図表、スキャンなどのサブルーチンが挙げられる。周知の如く、標準的なサブルーチンの利

用は、プログラミング時間の短縮やコンピュータ記憶の負担軽減のみならず、プログラミングの回避にもつながる。

あるコンピュータプログラマーたちは、標準的サブルーチンのみならず、どんな教師でも使えるような規格化されたプログラムも含めるよう提案して、さらに先の作業を進めている。「設問ベースの形成」支援システムという規格化プログラムは、設問ベースの創造、閲覧、保存、再現が可能である。具体的内容の蓄積は、通常、コンピュータ制御のものでおこなわれる。コンピュータはプログラム編集者に以下の順序で一連の動作を遂行するよう求める。

- 1) 第1問に先行するテキストを入力せよ
- 2) 第1問を入力せよ
- 3) 正解を入力せよ
- 4) 正答に対する反応を入力せよ
- 5) 第1問に対して予想される誤答群を入力せよ
- 6) 誤答に対する反応テキストを入力せよ
- 7) 解答時の試行回数を入力せよ
- 8) 最初の試行の後に与えられるヒントを入力せよ
- 9) 最終回での誤答に対する反応テキストを入力せよ
- 10) 第2問を入力せよ

このようなプログラムは、学課生成システムの名前で呼ばれている。生成システムの長所は、プログラミング経験のない教師でも「専用の」コンピュータコースウェアを作成することができる点にある。本システムの根本的欠陥は、予めプログラム化が完了しているという「硬直した」構造であることだ。とはいえ、多くの場合、たとえば、自動化された試験コースウェアを作ろうという時、生成システムはプログラム準備の総時間を短縮しつつ、教師に計り知れない助けを与えてくれる。

## 参考資料その2

「ロシア語のスペルチェック」について。出典は、  
КОМПЬЮТЕР, "финансы и статистика и COMPUTER", Москва, 1990, No. 3

辞書、辞書、……そしてスペルチェック

いわば戦場のような条件下でおこなう毎日の編集作業を根本から軽減してくれる2つのプログラムをモニターする機会に恵まれた。前者プログラム「КОРРЕКТОР」の作者は、Аркадий Борисович Борковскийである。販布をおこなっているのは、モスクワの「ТЕРМИНАЛ」社である。後者プログラム「ОРФО」を開発したのが、Олег Григорьевич Григорьев 主宰するモスクワ「ИНФОРМАТИК」社である。ОРФОのテストはバージョン1.0にておこなわれた、既にバージョン2.0が準備されているというので、この版の情報をも併せて提供したい。本誌「コンピュータ」編集部は二社に対し製品をテストする機会を与えてくれたことに感謝を述べたい。

どんな編集作業においても一番労力の要する「技術的」工程が、同一テキストの多回的書き換え (многократная перепечатка) である。我が編集部マイクロコンピュータが出現したとき、この作業労力は完全に減少したのだった。タイピストはもはや編集者や校正者が訂正したテキストを打ち直さなくてもよくなった。現在ではディスクに記録されたこれまで収録してきたテキストファイルを訂正するだけで充分なのだ。このおかげで、文書全体を書き換えるときには必ず必要とされた新たな打ち直し箇所が減少したのである。しかし、印刷テキストは、従来通り編集人の閲読に回さねばならない。

本稿の冒頭に挙げたプログラムは、本質的に、編集人になり替わるものである。プログラム課題は、テキストを「一読し」、誤りを指摘し、訂正の便宜を提供し、新しい単語の語形と意味を記憶することである。両システムに必須の部品は辞書で、その構成はテキストの校正に従って順次追加されていく。プログ

ラムは新しい語を個々のファイルに保存する。KOPPEKTOP は、新語をまるごと取り込むのに対し、OPΦO は語根のみである。

## KOPPEKTOP

両システムの最も大きい違いは、KOPPEKTOP が訂正箇所を含むテキスト全体を検査していくのに対し、OPΦO はディスプレイ画面に現れる部分のテキストしかチェックできないことである。また、テキストを修正するのにテキスト処理プログラム（エディタもしくはワープロのこと）を使わねばならないことだ。特に私のような出版関係の既にできあがっているファイルの作業をおこなう者にとっては OPΦO の短所となる。

OPΦO について手短かに語ることは不可能なので、読者には KOPPEKTOP から紹介しよう。このプログラムの手軽さは心を強く引くもので、毎日の作業でもこればかり使用している程だ。

KOPPEKTOP がロシア語テキスト（キリル文字コードが、選択的なバリエーションに一致していなければならない）を検査するには、まずプログラム名を、スペース記号のあとに、テキストファイル名を打ち込むだけでよい。以後の作業は KOPPEKTOP が自分で行ってくれる。テキストの検査中、プログラムが未知の単語を発見した場合、その語は文脈とともにディスプレイに出力される。今度はユーザーがその単語をどう処理するかを決定せねばならない。

- 1) 辞書に書き込む
- 2) パスする
- 3) 修正する

最後を選択した場合単語が専用の窓（ウインドウ）に現れ、カーソルをミスタイプ語に移動させて修正をおこなうことができる。修正後、プログラムはテキストの検査を続行する。「追加する」というオプションを選択した場合プログラムの知らなかった単語をユーザーの辞書ファイルに書き込む作業を自動的におこなう。「パスする」命令を選択した場合、KOPPEKTOP はテキストの検査を自動的に続行する。作業展開の状況（作業終了テキストのパーセンテージや

検査済みの単語数) については、ディスプレイの片隅に現れるデータが証明する。検査が終了すると、ユーザーには新しい単語を含む辞書と修正済みテキストをディスクに保存するかどうか聞いてくる。拡張子 DOC の更新は自動的におこなわれる。その後プログラムは作業を終了する。連続テキストの検査には上述の手順が繰り返される。簡単、速い、そして確実ということではないでしょうか。

KOPPEKTOP にはユーザーから見て一つの、それも本質的欠点がある。ユーザーの注意が絶えずモニター画面に釘付けになっていなければならない、実はこれが退屈極まるのだ。願わくば、テキスト中の未知の単語の提示にはプログラムが音声による信号を発してくれればと思う。そうなら、検査の途中でも作業を離れていることができる。これは重要なことである。KOPPEKTOP は単語の語尾変化をさせることができず、コンマの抜けた箇所を暗示しないし、また文体的な誤りも指摘し得ない。KOPPEKTOP は別の任務を帯びているのだ。つまり、各々の単語を辞書と比較すること、そして、もしそれに単語が見あたらなければ、ユーザーの注意を向けること、それだけのことだ。OPΦO に至っては事はもっと複雑だ。

### OPΦO

OPΦO とは常駐型の、即ちメインメモリー内に常に留まるプログラムのことである。これはテキスト（非グラフィック）モードを使用する任意のテキスト処理プログラム（エディタかワープロ）とともに稼働する。起動するにはまずコンピュータメモリーに読み込み、その後、エディタと検査にかけるテキストファイルを呼び出さねばならない。エディタ毎にテキストがディスプレイの異なる場所に配置されるので、作業の冒頭でプログラムに枠を指定しなくてはならない。その範囲にテキストが納められる。その後適切なキー配列を入力すると、OPΦO が起動する。このプログラムが作動する様子を紹介してみよう。作業の途中プログラムの知らない単語や呼応の不一致（文全体が計算されることがわかる）、さらに不正解な句読法を指摘しながら、カーソルがディスプレイ上

のテキストを「計算していく」。メモリーの余力があれば、自分のエディタで誤りを修正できる。もしなければ、他のキー操作によって、問題の単語を取り出し、そこにカーソルを移動させる。

OPΦO 1.0 はディスプレイ上に切り出されたテキストのみ取り扱うので、常時移動（スクロール）していなければならないので、特に、誤りの箇所が少ないテキストなどではかなり退屈である。

私の好きなエディタは PC-Write である。OPΦO にてテキストを検査するために私は校正プログラム、それからエディタ、最後にテキストファイルをロードしなければならない。一連のキー操作をおこなって、テキストの検査がおこなわれるフィールドの境界を定める。一定のキー操作をおこなうと、カーソルは未知の（間違いの？）単語に置かれる。この単語についてミスタイピングであれば、エディタを用いてそれを修正する。すべて正しくても、OPΦO にその単語が登録されていなければ、再びキー操作をおこない。新しい単語を辞書に書き込む準備に取りかかることができる。これは普段行われそうにない作業である。プログラムはカーソルに示された単語の検討を始める。ユーザーはまずその単語がどの品詞に属するかに答えなければならない。メニューに従って、カーソルの移動と当該の単語に対応する品詞を示すことによっていくつかある可能性のうちから一つが選択される。その後さらに 3, 4 つ程の質問が課せられる。それに答えながら、ユーザーは新しい単語の 3, 4 つの語形をシステムに指示していく。質問がどのようなものかは、確認を受ける単語とユーザーの回答次第である。ロシア語の知識が不十分という最悪の場合には、質問に対して何度か答えさせられるはめになる。というのもシステムが不正解な回答の受取を拒否するからだ。もっとも複雑な場合私は 7 つの質問に答えなければならなかった。幸いに、判断の全過程を通じ、キー一つで説明（ヘルプ）を呼び出し、必要な情報を得ることができる。

ユーザーが新しい単語を辞書に書き込む前に、OPΦO がユーザーの質問をもとに未知の単語の語形すべてを作り上げ、チェック用にそれを切り出すのである。ユーザーが誤りを犯したら、質疑応答の手順を繰り返さねばならない。全

て正しければ、「入力」を押すだけでよい。

一切合財がめんどうと思われたのであれば、それは筆者の紹介の仕方が悪い。敢えてこのプログラムの作業手順を手短に、単純に説明しようと考えなかった。KOPPEKTOP に比べ、このプログラムのもつ優位点は、質疑応答モードの後辞書には一語でなく、一時に数語、(名詞であれば) 12 語から、(動詞なら) 20 語までの登録が可能なことである。

このソフトウェアを使用するととても簡単に、且つ迅速に作業が進むという評価を読者から頂いている。本稿を結ぶにあたり、いくつかの時間的計測値を示しておきたい。本稿のチェックには KOPPEKTOP 上では(プログラムの読み込み開始から DOS への復帰まで)7.5 分費やした。この間に 14 箇所の誤りを修正した。辞書には 128 語が書き加えられ、修正テキストは今後の作業に使用し得る。この作業で筆者は 270 回キーを打った。検査を受けたテキストファイルの長さは、10,342 バイトである。検査語数は 1,333 語であった。

同じ作業を ОРФО の環境で実行すると、(プログラムロードから修正済みのテキストファイルをディスクに戻すまで)10.5 分を必要とした。14 箇所の誤りが修正され、5 回の質疑応答で 58 語が辞書に追加され(ОРФО の辞書サイズは KOPPEKTOP のそれに比べ著しく大きい)、2 つの文体的な誤りが整備された。キー操作は 350 回に及んだ。

ОРФО バージョン 2.0 では、筆者の確信するところ、テキスト修正の速さが約 1.5 倍向上している。その他、プログラムは新しいスペックをもつ。

1) 誤った単語を正しいものに自動修正

2) ハイフンを伴う単語の一致を検査

(例; 「女医 женщиной-врач」は誤りとして点滅)

また、ある種の接頭辞との語形成上の正しさを検査

(例; поляблокаはпол-яблока と修正される)

3) 自動チェックモード、誤った単語が検出された場合のシステムが音声信号を発する

この種のプログラムは、西側では spell-checkers と呼ばれている。それはか

なり普及しているが、通常はエディタ（ワープロ）に内臓されている。ポーランドには、同じ原理のテキスト処理機能をもつ同種プログラムが2種（TAG と QRTEKST）がある。ОРФО は、КОРРЕКТОР のような通常の spell-checkers の枠をはるかに超えているし、知的（интеллектуальные программы）修正プログラムに分類が可能かも知れない。

ソビエトのプログラマの根底にある発想は、非常に魅力的なものだ。特に出版業界に職業人として従事する者にとっては、理想的な完成像は КОРРЕКТОР の簡潔さと ОРФО の言語学的仕上がりを結合したものであらねばならない。また、テキスト全体にユーザーが選択するコード付き句読点を振り分けていく能力を与えるものでなければならない。この種のプログラムを手がける者なら、コンピュータを利用したロシア語通信の現場でならどこでも販路が見込めるであろう。

仕様一覧		
プログラム名	КОРРЕКТОР 1.0	ОРФО 1.0
著者	Аркадий Борковский	Олег Григорьевич 代表
使用言語	C	C, アセンブラー
内容	ディスク 3 枚 5.25 インチ 2 DD	ディスク 3 枚 5.25 インチ 2 DD
マニュアル	ディスク上に記述したファイル	小冊子
価格	1 セット 1,400 ルーブル	1 セット 1,100 ルーブル

## 追 記

本稿は、平成元年7月決定の札幌大学研究助成による成果の一部である。当初ソ連で出版されると聞いたロシア語コースウェア情報にしたがって、コンピュータ学習の実例を紹介しようと考えていた。当時の書籍カタログには次の

ように予告された。

『コンピュータによるロシア語百題選』(新刊情報, 1989 年第 6 号)

参考書は問題全部と指導が納められている 2 枚のフロッピーディスク, それに小冊子類からなる。小冊子にはフロッピーディスクの指導方法が網羅されている。

本書には様々の問題に含まれる複雑, 且つ多岐に亙る言語資料のマスターとブラッシュアップを目指す。様々のロシア語レベルにある学生のための, また, パーソナルコンピュータを所有する独学者のための, 教科書補助教材としての利用が可能である。中級程度のレベル, 1990 年第 4 半期出版予定, 価格 21 ルーブリ, 2 万部, 『ロシア語』出版社

以来 2 年余りが経過したが, この出版計画については何も聞こえてこない。その間米国では独習用「ロシア語 一步一步」が米ソ共同作業で完成している。今回の報告には間に合わなかったが, 冒頭に示した 7 段階の発展スケールにソビエトも無頓着に跡付けているとは信じがたい。確かに, 教師のための参考書には陳腐な記述や既に時代遅れの感がする手法を盛り込んでいるが, 一方で, 新しいものの萌芽も確実に認められる。オーサリングシステムの構築やプログラムのモジュール化, マルチメディアの提唱が同一の書物に述べられる。学者が退屈しないように, 学習者に無用の苦痛や不自由を与えないように, また, 願わくば学習者の「反問」を受けて立ってくれるようなそんな関係を期待しているのだ。それにはまず学習ソフトウェアといえど学習者の入力内容を認知する能力の開発から始めなくてはならないだろう。語学学習にはシュミレーションや試行錯誤の側面が, 例えば理科や数学に比べ, 少ないのかも知れないが, 学習者が協見することを許さない, 出力は音声と貧弱な画像だけ, しかも学習者からの入力手段はキーボードのみの CAI コンピュータなら「不要だ」と言い切れる環境を実現したい。これは現に日本国内の状況でも, また上述書からも導き出すことのできる結論である。