

7L-3

計算機モデルを用いた  
パソコン言語訓練システム

中山康子, 会田一夫  
株式会社東芝 総合研究所

1. まえがき

CAI は、従来のフレーム型と、人工知能の応用として最近注目されているICAIに大別される。フレーム型CAI では、教材プログラムは複数のプログラム・フレームから構成され、フレーム内に教材(教育対象の専門知識)と教育手順を混在して記述するため、学習者は質問の形で専門知識の参照を行えない。それに対し、ICAIでは専門知識と教育手順とを独立して記述するので専門知識の参照が可能となる。しかし、専門知識をIF~THEN形式で断片的に記述する従来のルールベース型CAI では、基本的な原理/原則(深い知識)は表現できず、そこから得られる現象/結果(浅い知識)のみを表現する。このため、事前に用意した浅い知識以外には学習者からの問合せに対応できず、対応の柔軟性に問題がある。

パソコン言語訓練システムはこのような背景のもとに開発するもので、初心者を対象としてコンピュータの基本原則とプログラミングの基礎知識を教育するための、モデルベース型CAI である。モデルベース型CAI は、深い知識までも対象としてモデル化し、そのモデルが学習者に回答する。したがって、学習者からの多面的な問合せに柔軟に対応できる。

2. システム構成

本システムは、パソコン初心者がBASIC 言語を習得するのを支援するものである。システム構成を図1に示す。教材データベースは、BASIC 言語の教材(例題プログラム、穴埋め形式の練習問題)を蓄える。学習者モジュールは、学習経過の記録と、学習者の理解度評価を行なう。計算機モデルは、システムあるいは学習者の指示によりBASIC プログラムを実行する。システム全体を制御する指導戦略モジュールは、学習者の理解度と教材の事前条件(事前に学習すべき教材内容)を参照して、次に提示すべき教材を選択/実行する。

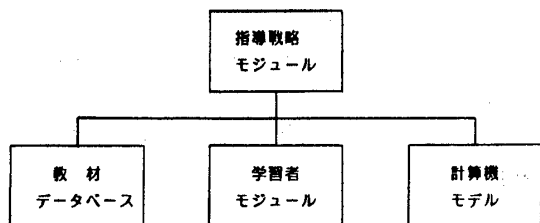


図1. システム構成

3. 計算機モデル

BASIC は初心者用プログラミング言語として最も広く普及しているが、言語の文法はコンピュータ特有の概念で構成されており、コンピュータに関する知識なしに理解することは困難である。例えば、変数、繰返し等の抽象概念、あるいはコンピュータ特有の算術式 LET S=S+1の理解には、コンピュータの内部構造やプログラム実行に伴う内部の状態変化、つまりコンピュータの「からくり」の理解が必要である。「からくり」は構成要素の物理的/空間的な関係と動作であり、この種の知識は記号表現より図形/動画表現の方が理解しやすい。そこで、従来 black box化されていたコンピュータの「からくり」をモデル化し、ビジュアルに表示する計算機モデルを導入した。図2は計算機モデルの表示例で、「コンピュータは、入力装置、出力装置、演算器、プログラム、プログラム・カウンタ、命令解釈器、メモリの部品から構成され、これらは部品データベースによって接続されている」ことを表す。学習者は、計算機モデルとの対話によりBASIC 言語を習得する。計算機モデルはプログラムを実行しながら、コンピュータ内部の動作を学習者に説明する。

また、学習者が入力した誤りのあるプログラムに対しても計算機モデルはその実行過程を動画表示し、所望の結果が得られないことを学習者に確認させる。

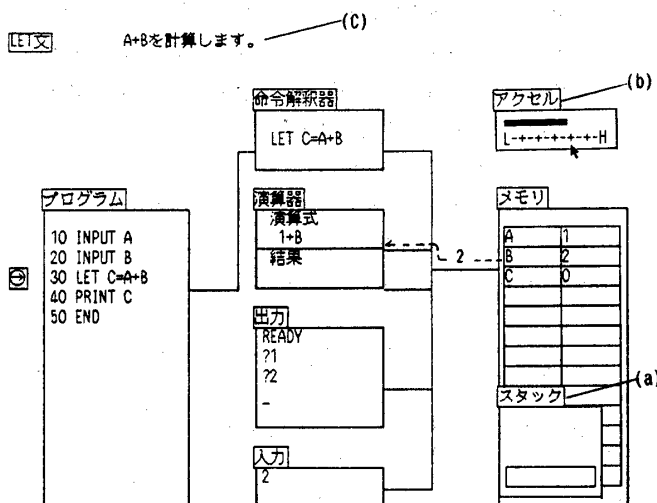


図2. 計算機モデルの画面表示例

#### 4. 計算機モデルの構成

モデル構築のポイント、表現する知識のレベルと構成要素の選択である。BASIC プログラミングの知識は、ハードウェアに近い順で以下に示す6レベルに分類できる。

- (1) 機械語：計算機各々の機械語のsyntaxとsemantics
- (2) 各計算機に共通の基本機能：一連の機械語に対応した

計算機特有の言葉

(例) LOAD A ; ADD B等

- (3) 文：BASIC文のsyntaxと、上記基本機能の組合せで表現されるsemantics

(例) LET C=10    LOAD 10 , STORE C  
       LET C=A+B    LOAD A , ADD B , STORE C

- (4) 構文：繰返し、分岐、飛越し等の構文知識
- (5) パターン：文の定型的な組合せパターンに関する知識

(例)「ループによる計算は、初期設定、計算の本体、ループの脱出から構成される」

- (6) プログラム：プログラムの設計に関する知識等

計算機モデルで表現する知識はレベル2～4である。レベル3の知識を理解するには、より詳細な計算機の基本原則に関する知識を必要とするが、機械語レベルの知識はBASIC 初心者には詳細すぎる。そこで、例えば変数の概念の説明には、16進数の番地ではなく、レベル2に相当する「メモリ」(図2参照)を導入した。また、レベル4の知識についても、その理解には基本原則の知識を必要とする。例えば、計算機モデルはFOR ~NEXT文の制御機構を、「プログラム・カウンタ」を用いて次のように説明する。まず FOR文に対応するNEXT文の捜査により実行範囲を捜す様子を示し、次にこの範囲を行きつ/戻りつ実行することを説明する。図3(a)は、第20行のFOR 文に対応するNEXT文を捜査している状態を示す。「プログラム・カウンタ」シンボル部の下向き(↓)表示と、ボックス部の延長でNEXT文を捜査している様子を表し、NEXT文を発見した所までが繰返しの範囲となる。図3(b)は、シンボル部がこの範囲を行きつ(↓)表示/戻りつ(↑)表示)実行する様子の動画表示である。なお現在実行中の文は、シンボル部を(→)表示する。FOR ~NEXT文のバランス・エラーの説明を図3(c)に示す。NEXT文を発見できなかったことを、「プ

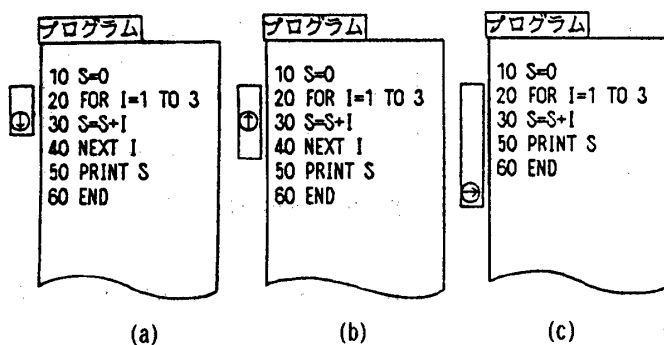


図3. FOR ~NEXT文の「からくり」の説明

ログラム・カウンタ」のボックス部が最終行まで延長し続けることで説明する。また、GOSUB ~RETURN文については、「スタック」(図2(a))を導入して説明する。

レベル5以上は、プログラミングそのものに関する知識であり、ソフトウェア開発モデルといった別の知識体系を必要とする。

#### 5. 計算機モデルの機能

- (1) 基本原理のビジュアル表示

計算機モデルは、オブジェクト指向で構築されている。計算機システムを構成する各部品の実体に対応するオブジェクトを定義し、計算機モデルはその集合として構成される。計算機モデルの各オブジェクトは、その処理機能をビジュアルに表示するために、図形情報と動画手続きをもっている。これらは計算機モデルの処理機能実行時にオブジェクトが起動されることにより読み出され、画面表示される。

- (2) オブジェクトの表示/非表示

計算機モデルの構成要素は4で述べたレベル2～4の知識を表現し、学習者はレベルを追って訓練を行なう。ところが、全部の構成要素を常時画面に表示すると、学習者は必要以上の詳細な知識を参照することになり、思考の混乱を生じる。そこで、計算機モデルの構成要素に対して表示/非表示(画面に表示しない)状態を設定できるようにした。

- (3) 実行速度調節

計算機モデルでは実行速度調節機能を提供し、学習者の要求に対応できるようにしている。実行速度調節を行なうために、「アクセル」を呼び出したものが図2(b)である。

- (4) 説明文の生成

計算機モデルは、随時適切な説明文を生成し画面表示する(図2(c))。説明文の生成は、オブジェクト毎の説明用テンプレートを読み出すことで実行する。

- (5) オブジェクトの操作による教材作成

計算機モデルをオブジェクト指向で構築することにより、以下のようなモデルの操作上の利点があり、学習者のレベルに応じた教材作成を支援する。

- ・オブジェクトは単体で動くので、コンピュータ内部の機能を個々に説明できる。
- ・オブジェクトの表示・非表示の指定により、学習者の注目を集中できる。
- ・学習者のレベルアップによる新しい部品の追加や置換えが容易にできる。

#### 6. あとがき

計算機モデルの開発を終了し、現在、教材シナリオの整備と計算機モデルの評価を行っている。今後は、学習者の理解度に応じた教材選択あるいは典型的な誤りの抽出を支援するための学習者モデルの開発に注力する。