

シミュレーション型情報処理教育CAI におけるコースウェア作成上の問題点

伏 見 論

情 報 数 理 研 究 所

パソコンを利用しスタンドアロンタイプの情報処理教育CAIで、CAI専用の言語プロセッサを内蔵したコースウェアの作成事例について述べる。その作成意図の紹介と、今後改良すべき点などについての現状把握を述べ、インテリジェントなシミュレータのCAIへの組み込みによるコースウェアの展開法を採る。

A Practical Discussion on Simulation-type Courseware Development
for Computer Programming Course CAI

Satoshi FUSHIMI

Information and Mathematical Science Lab. Inc.

1-55-8, Yoyogi, Shibuya-ku, Tokyo, 151 JAPAN

"IMS CAST BASIC" is a CAI courseware for computer programming education and have a built-in BASIC interpreter with some CAI-tuned features. The author discuss educational intent of that courseware and how to utilize effectively an intelligent interpreter/simulator for CAI.

1. CAST BASIC システム

情報処理教育をC A Iその他、計算機環境をうまく援用して行おうとする試みは色々と試みられているが、ここに紹介するのは、私どもで開発した主として一種のシミュレーション的な考え方を軸としたC A Iシステム上のコースウェアについての、開発における留意点および反省点などの列挙である。一つの事例紹介的なものとして理解いただきたい。

ソフトウェアとしてのシステム名はCAST BASICといい¹⁾、オーサリングシステムなどの汎用C A Iシステムに対応した機能をほぼ持っているが、ソフト全体がこのコースのために設計・製作されたものである。「実行」時の機能的特色：

- ①教育用としての機能をもったB A S I Cインタプリタの内蔵
- ②いわゆるエグゼキュータとこのインタプリタとの連携動作
- ③C A Iの教材画面を対象とした検索方式によるデータベース機能の提供
- ④学習結果の評価において、インストラクタやシステム管理者のレベルで書換え可能なルール形式(個々の問題画面の属性に基づく判定および上記の①による実習結果に基づく判定)の学習履歴評価システムの装備

2. 想定する学習環境とレディネス

このシステムは、基礎的な情報処理教育の内容をできるだけ実機環境を利用しながら実践的に呈示しようとするもので、いわゆる「手に覚えさせる」という感覚をねらったものと要約することもある。想定している学習者は、プロの情報処理技術者ではなく一般情報処理教育におけるプログラミングの基礎的な理解/技能体得をめざしたものである。マニアックなユーザは対象でない。

学習者のレディネスとしては、高卒程度で、B A S I Cの予約語群の原義は一応わかり、三角関数などの数値関数の意味も一応わかるべきものとして本人が行動してくれる程度のことを想定している。ハードウェアの概念やファイル概念などは通常の社会常識以上のものは特に想定していない。

3. プログラム言語学習・教育についての諸先例にふれて

プログラム言語教育は全体としてもいわゆる文法教育オンリーからの脱皮が叫ばれて久しいが、そのC A Iにおいてもその傾向は現在進行形で進んでいると考えられる。B A S I Cの教育をとってみても、コマンド仕様の解説文の羅列とその説明が理解できたか否かのテストというスタイルから離れ、いろいろな意味でプログラミング実践に近いものへと変化してきていると考えられる。

・CAST BASICのシステム構想においては、UNIXにおけるLEARNプログラム²⁾を一つの基準として考え、さらに「言語教育ではやはり系統的な概念の呈示は欠かせない」こと、および、学習者には常に何か課題作業をさせるがその作業過程自身の中でのアドバイスや学習者側からの質問に答える機能は欠かせないことを考えた。教室での実習の支援のためのシステムとしては磯本等による報告³⁾のようになんとかのツールを組み合せ全体の進行はインストラクタが指導するタイプのものも多いが、われわれのシステムではスタンドアロンの製品という性格からインストラクタがそばにいないとは前提にできないという事情もあり、学習者のある程度自主的な実習過程をいかに丁寧にサポートするかが問題となると考えた。

・B A S I Cのプログラミングにおける誤答の解析などについては、すでに対

馬等のシステム(BSADV, BSDIAG)⁴⁾があるが、CAST BASICではインタプリタそれ自体がプログラムの構造を解析しながら誤り指摘をするという方法を用いている。

4. コースウェア作成の方針

CAST BASICでは、「言語」教育の目的は、本来の汎用的な基盤上での処理の組立の理解と実践にあると考える。その他次の点に配慮した。

- ① 構造化プログラミング的なコーディングスタイルの立場をとる(インタプリタによるスタイルチェックもおこなう)。また、言語の文法教育が中心というのではなく、操作法、用例、状況例、コンピュータ世界の種々の概念など総合的にプログラミング手法/設計法として問題になる事柄を呈示/実習するようにする。
- ② 基本的なコースの進め方として、短い解説や導入に基づきそれに対応する<実習>を繰り返す、というパターンをする。<実習>は画一的な課題/正答例を打ち込めればよいのではなく工夫して行動することを奨励する。また、コースの進捗についても学習者の自主的な選択の幅を増すため、完全にメニューからの選択方式をとる。ただし、コース自体には明確なシナリオがあり、個々の教授内容も明確に規定された事柄を扱うようにする。
- ③ 学習者に対する評価はエグゼキュータのいわゆるKR、インタプリタのメッセージ、総合評価システムによるルール評価の3つの組み合わせによるので、それに付随する評価用データを準備する。この評価用データも教授内容と理解する。
- ④ BASIC学習において初心者理解されにくい点には重点的なコース配分を行いシミュレーション(アニメーション)的な画面を用いる。
- ⑤ ②とも関連するが、例文プログラムを多用し、また短いプログラムだけでなく実用規模のプログラムも参考のために大量に呈示する。
- ⑥ 学習者の多面的な「質問」に応じられるようにする。

このうち、①については、構造化されたBASIC⁶⁾を用いるのがソフトウェア作成としてはやりやすいが、市販用のCAIソフトとしてはユーザの多いものに密着せざるを得ないのでN88BASICの仕様のプログラムを構造的にインタプリタ側で読み替えるということにした(いわばN88BASICのシミュレータ)。②はこのコースウェアの重点的な特色とした。③は「評価」といってもCAIから学習者へのダイナミックな会話の総体を問題にしたいので、考え方としてはAI的な手法の導入や誤答に対しての教授法を深く掘り下げるべきテーマだが今回は上記のような組み合わせで行われている。④の点については短大初年級クラスのエデュエーション経験者の意見を取り入れたほかPutnum等による報告を参照した⁷⁾。

これらの方針の考察においては、たとえば菅井の呈示するような学習環境論⁸⁾をふまえることが必要だろうがわれわれの現実はそこまでは意識できていない。

5. 方針の実現方法

①のためにソフトウェアとしてはLS-BASIC(Learner's Structured -)と称する専用インタプリタを開発し、またコース内容としては「構造」「設計」の立場からプログラミングを呈示するという方向をとる。またユーザーとして要求仕様からプログラム設計に落としていく過程についての解説を加える。

②のために、解説画面、実習画面の組み合わせと「確認」のための演習画面というスタイルを定型的なものとして持つ。(なお、演習画面の穴埋め判定でもその回答内容をBASICの文法や出題意図に照らして等価なものは回答例にマッチしたものとみなすオプションを用意する。)

シナリオの軸としては、「試験の成績処理」というテーマに沿った500行程度のプログラム例を逐次解説展開していくというストーリーを持ち込み、それに教授上のシナリオを絡ませていく。

③のKRは正解例、誤答例など複数のものについてすべてのフレームごとに独自のものを検討し、用意する。LS-BASICのメッセージは、基本的にはエラーメッセージの拡張という体裁である(∴限界がある)が、できるだけ原因サイドに立ったものとする。また、このエラー判定はその履歴をエグゼキュータに引渡し次の総合評価システムで総合判定に利用する。総合評価システムは演習問題の可否と上記のLS-BASICのエラー履歴をもとに判定を行うがそれに個々の演習画面の意味的な情報を伝えるため画面ごとにその意味的な情報をコード化した属性データをもたせる。

④については、LS-BASIC自身に変数値の実行時変化をトレースする機能をもたせる、これとは別にデータとしてもたせたBASICプログラムテキストを逐次実行しながらその実行状態(実行行、変数値変化、ユーザーI/O)をマルチウインドウ風画面にビジュアルなトレースアニメーションする機能を用意する。

⑤については、個々のコースウェアの作り方のほか、合計1万6000行のコメントつきBASICプログラム群を用意した(実はあまり使われていない)。

⑥については、検索システムでオンラインマニュアルや他のCAI画面を参照できるようにするが、そのために関連用語のシソーラスを用意する(逆引用)。

6. 大局的な組立

コースウェアは約1200画面で全体の目次の組立ては次のようになっている。

序章	このシステムの使用法と前提知識および参考知識の解説的呈示
1章	BASICの入門的な一通りの知識と実習事項 (このシステムの中心部分)
2章	プログラムの設計法として要求仕様から設計書やプログラムにおとすまでのコーチとプログラム開発手順(デバッグなど)
3章	プログラミング各論として、入出力などの分野ごとに定石的なプログラミング法を示す
(4章)	応用プログラムのコーディング例(コメントつきだが、説明抜き)
(5章)	BASICオンラインマニュアル(検索システムでの利用とコースウェア中でのマニュアル参照練習に用いる)

7. 個別テーマ

認知の進行についての方針: BASICは最初CAI的な目的を持って開発されて以来「インクレメンタルに利用できる」(Kemeney)という良さを持っているということはある程度できるだろう。コース中で繰り返し行う<実習>の展開においてもこの側面を活用するようにしたが、構造化プログラミングの視点を

強調するために、例文は最初からストアされたプログラムとして呈示しダイレクトモードは副次的にしか扱っていない。単純でオプション利用の少ない構文例を用い、実行／書換え／作成／バグとりなどの様式で、つぎのようなテーマの展開をとっている（1章）。

プログラムというもの → 数値の出力法 → 単純な計算 → 入力とそのエラー判定 → 反復と配列 → サブルーチン → 関数 → フォーマット出力 → プログラムソースの保存など → ファイル出力（順編成） → プログラム書法 → 入力の諸手法 → 出力の諸手法 → 式と演算の文法

ループの種別として前判定・後判定・中央判定という内容を扱うコースを当初は提供していたが、内容が重くなりすぎるという反省に立って、ループはWHILEループが基本だとして教えることでここではあまり深入りしないことにした。

このほかに、検索システムやオンラインマニュアルの参照も意図的に課題として行わせている。一般に、インタプリタの利用やマニュアルの参照などは、自由課題のような形で「自由にやらせる」というスタイルを大幅に採用している。

操作技能の進歩についての方針： プログラムのメタ的な取扱い（編集など）については次のような持っていき方をしている。

リストの表示 → 実行とその結果の参照 → 既成プログラムの一部変更 → 画面上での編集操作等 → 小さい自作プログラム → 大きいプログラムの部分（モジュール）の作成 → デバッグその他のための操作
この際にBASICのマニアックなユーザのプログラミングの方法としてトライアンドエラーを繰り返し論理抜きに経験的にノウハウを獲得していくということがみられる。自主性の面で良い面もあるが、本来的な情報処理教育ではこれは好ましくなく、論理的な設計と動作の論理的な理解を主眼にしたいと考えた⁷⁾。

「構造」のテーマについてのコース内容： ループの意味や選択実行についてはしばしば初心者が意味を取り違える／理解できない事柄となっている。LS-BASICは「構造」が正しいかどうかのチェックをかなり行う。また、この点については特にビジュアルシミュレーションを用いて、コントロールの動きと変数値の変化を連動させて示すなど視覚的印象によって正しい理解へ持ち込むように考えた。

入出力のテーマについてのコース内容： 会話的入力の意味についてもある程度「動作シミュレータ」によるとともに、いろいろな入力／出力様式について実験して確かめるようにさせている。また、全体のストーリーである例題プログラムはメニュー選択処理やデータ入出力処理中心でそのためのプログラミングのありかたを学習者とともに解析するというスタイルをとっている。

宣言や変数の型についてのコース内容： 基本的にこれらは2、3回に時期をわけて次第に厳密に教えるスタイルである。LS-BASICは型違反などに対してはかなり有効なメッセージをだすので学習者にいろいろと実験をさせてみるができる。ただし、LS-BASIC自体は整数型の数値型しか持たないため限界があり、改善されなければならないと思っている。

出題意図との関係でのメッセージ： LS-BASICのメッセージ機能はかなりうまく働いていて、例題作成にも重宝したぐらいであるが、出題意図を反映した判定／メッセージ機能はない。この点は、完全なインタプリタとしての機能を損なわないようにしながら改善する必要がある。

コースの自動・手動組替えについて: いわゆるKR情報は学習者モデルに基づいたものでないとあまり的を射たものになりにくい、製品としてまとめる上では学習者モデルと言い切れるものを組み込むことは断念し、メッセージの内容の工夫と総合判定機能(上記参照。主としてデータの統計処理の指定と判定からなる)で評語を与えることにした。しかし、せっかくの判定結果を活かす方法が内在していない。なんらかのコース組替え機構との結合が必要と考えている。

8. 結 論

・CAST BASICのようにある程度インストラクタ抜きの状態を想定するシステムでは機能的には(通常頻繁に使われる機能に加えて少ししか使われなくても)豊富な機能を持ち、また(直接学習者に呈示しない可能性があっても)非常に豊富な知識内容をシステムが持ち、必要なときにはそれらが呈示できる、というのが望ましい。コンパクトにまとまっているが奥は非常に深いというのが理想的だろう。この意味で見て、1200画面程度プラスLS-BASICでもかなり不十分である。

・情報処理教育においてCAI用の言語プロセッサ/シミュレータやCASEツールのようなものを内蔵するという形態はその自然さゆえに今後ほとんど不可欠なものとなると考えている⁹⁾。その場合、CAIのコースウェアづくりも非常に多様な要素(情報化社会論や実務教育の学習環境論などもふくむ)をますます強く考慮せざるをえなくなるので、それに見合ったコースウェア作成のノウハウ/理論形成が今後急がれるという気がする。

・情報処理教育でのコンピュータの役割を①学習者のツール②稼動する学習対象システム③それらを媒介する環境/筋道/動機付けとしてのコースと分けた場合、ある程度①を重視してみたわけだがこれらの総合的な設計が必要だろう。

[参 考 文 献]

- 1) 伏見: 新しいシミュレーションCAIをめざしたCASTシステムの特色, 教育情報研究, Vol 2, No 4, pp55-58(1987)
- 2) Kernighan, B.W., Lesk, M.E.: LEARN-Computer-Aided Instruction on UNIX(Second Edition), UNIX PROGRAMMER'S MANUAL(Seventh Ed.), Bell Laboratories, 1979
- 3) 磯本, 小島, 木村, 石桁: FORTRANプログラミング演習授業支援のためのCAI, CAI学会誌, Vol 5, No 1, pp45-57(1987)
- 4) 対馬, 加賀: BASIC誤答診断システムBSDIAGの戦略の改善, 第12回CAI学会研究発表大会論文集, pp109-115(1987)
- 5) 菅井勝雄: CAI利用のための学習環境論の構築とその課題, 第11回CAI学会研究発表大会論文集, pp73-76(1986)
- 6) Kemeny, J.G., Kurtz, T.E., Elliott, B., et al.: True BASIC Reference Manual, Addison-Wesley, 1985
- 7) Putnum, R.T., Sleeman, D., Baxter, J.A., Kupsa, L.K.: A Summary of Misconceptions of High School BASIC Programmers, J.Educational Computing Research, Vol 2(4), pp459-472(1986)
- 8) 西田, 佐野, 鷹津: CD-ROM CAIシステムの研究I, 教育情報研究, Vol 3, No 4, pp4-13(1988)