

## 3H-3

HyperCardに基づく  
プログラミング言語用知的CAIシステム

酒井 仁、海尻賢二

(信州大学)

## 1 はじめに

CAIにおいてユーザーインターフェースは、コンピュータが人間の教師の役割をすることから、学習者にとって自然で優れたものが要求される<sup>1)</sup>。そこで、ユーザーインターフェースとしてHyperTextの概念を実現したツールであるApple社のHyperCardを用いてプログラミング言語用知的CAIシステムを作成する。このことにより、以下のようなメリットがある。

- ・豊富なリンク機能により、学習者は自由に教材知識をたどることができる
- ・解説が整理され格納されているので画面が見やすく、思考パターンが妨げられにくい
- ・ビジュアルな動作によりユーザーのモチベーションを高めることができる<sup>2)</sup>

## 2 システムの概要

このシステムはプログラミングの経験がほとんど或は全く無いものを対象とする。プログラミング言語としてはPascalを使用する。

複数のユーザーに対応できるように、各学習者ごとに進行度、理解度を記録した履歴のスタックを用意する。これを随時参照することにより個人個人に応じた教授指導を与える。

システムの全体構成を図1に示す。

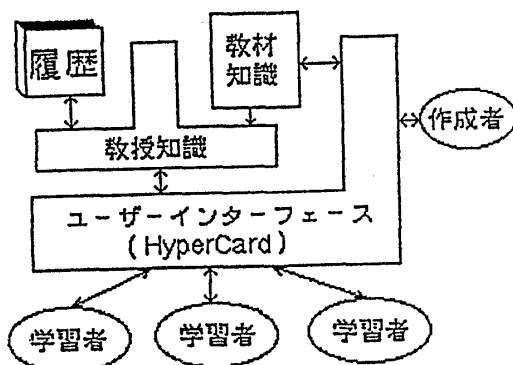


図1 知的CAIの基本構成図<sup>3)</sup>

## 3 教材知識の構成

HyperCardは、一画面に常に一つだけ現れるカードを基本構成要素とする。カードの上にはフィールド、ボタンなどのオブジェクトを付加することができる。また、複数のカードが集まってひとまとまりになったものをスタックという。

このシステムでは、教授すべき知識全体を<for文><配列>などの15の単元に分け、それぞれスタックを割り当てる。それら一つ一つは次のデータを有する。

- ・スタックの名前
- ・解説文(日本語)

解説文中のキーワードには更なる説明がリンクされており、学習者はそれを自由に掘り下げることができる。

## c・例題

例題はアニメーション的な実行ができ、プログラムの流れや変数の値の変化が視覚的にわかるようになっている。

## d・構文図

構文図はBNFとブロック図によって記述されており、階層的にリンクが張られている。

## e・関連するスタックへのリンク

各スタックはその単元と関連するスタック(<repeat文>と<while文>など)とリンクされており、必要に応じてナビゲートできる。

## f・問題

各スタック毎にいくつかの形式で設問が用意されている。ここで得られた解答により、システムは履歴と相談して学習者の理解度を判断し、次の指導方針を決定する。

## 4 問題による教授戦略

問題には次のような形式がある。

- ・選択式問題  
yes/noまたは記号で選択させる
- ・入力式問題  
解答自体を入力させる

- 穴埋め式問題

非常に小さなプログラム単位を入力させる

- プログラム演習

練習問題を与え、他のアプリケーション上で数十ステップのプログラムを作成させる

選択式問題、入力式問題、穴埋め式問題にはそれぞれカードを数枚割り当て、さらにその上に数枚の問題フィールド載せる。一枚のカードの上には同種の問題フィールド（数値など一部だけが異なるもの）を載せ、教授戦略によりどれか一枚だけ表示する（図2）。問題フィールドは次のデータを有する。

- 予想される解答のリスト

• それらの解答が選択されたときに調整を行う理解度変数のリスト

- そのとき各理解度変数にかける重み  $\alpha$  ( $0 < \alpha \leq 1$ )

理解度変数とは、各単元に設けられたチェックポイントの理解の度合を0~1の数値で表したものである。初期値は0で、そのチェックポイントの解説を表示した時点で1になる。その後問題によって調整される。これはどこまで進んだかを表す進行度変数と共に履歴スタックに記録され、ユーザー登録の際にグローバル変数に読み込まれる。

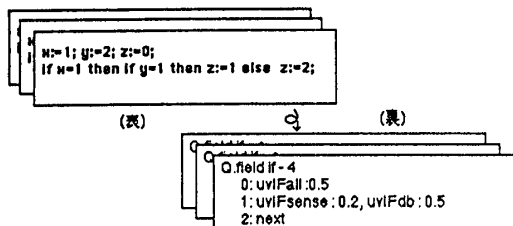
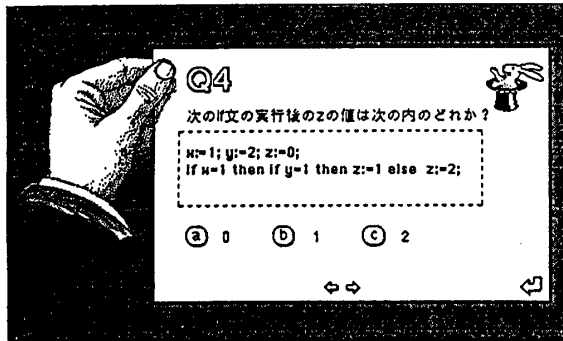


図2 選択式問題カードと問題フィールド

## 5 システム全体の流れ

まずシステム起動時にユーザー登録を行い、履歴スタックとの対応付けを行う。その後そのユーザーの進行状態を表示したメニュー画面を表示し、復習をするか先に進むかを選択させ、指定された単元のスタックへ移行する。

一つの単元スタックに置いて、ユーザーはメニュー

画面（図3）のボタンを押すことにより解説、構文図、例題を自由にブラウジングすることができる。但し問題にはいるには、以上の全てのカードを一回以上開いた後でなければならない。（一度開いたカードはメニュー画面上で右肩に小さく表示する）

問題においてはフローはシステム主導で行われる。まず各チェックポイントの理解を問うような問題を選択式及び入力式の問題カードで出題する。正答が返ってきたならば次の問題カードに移る。誤答の場合、その問題カード上の問題フィールドの持つ解答パターンと照合し、各理解度変数の値を調整した後同じカード上に新たな問題フィールドを表示する。

理解度変数が更新された際にはいつもその値が負になっていないかのチェックが行われる。もし負になっていた場合、システムは問題を中断し、メッセージを表示した後その理解度変数に対応する解説のカードに強制的にナビゲートし、そこでユーザーに理解を促す。その後、ユーザーが再び問題に戻ったときには直前の問題から開始できるようにする。

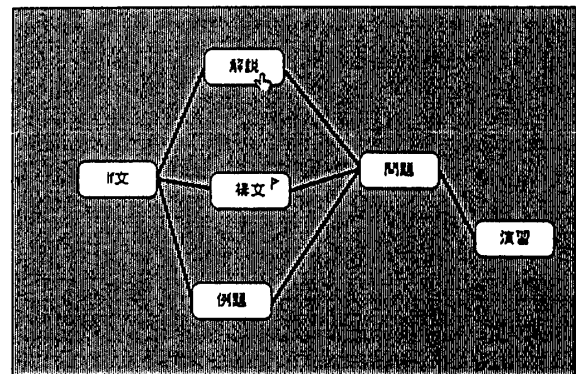


図3 メニュー画面

## 6 おわりに

本稿では、HyperCardを用いることにより、柔軟性のあるCAIシステムが構築できる事を紹介した。これは現在Mac II上でHyperTalk (HyperCard制御用専用言語) で作成中である。

## 参考文献

- 1) 溝口 理一郎他「知的インターフェース」  
電子通信学会誌 Vol.71 No.4
- 2) Danny Goodman 「THE COMPLETE  
HYPERCARD HANDBOOK vol.1」
- 3) 大槻 説乎他「知的CAIのパラダイムと実現  
環境」 情報処理Vol.29 No.11