

CAIソフトの作り方

愛知県立千種高等学校	たかはら 高原	ふみのり 文規
愛知県立国府高等学校	とりい 鳥居	ひでき 秀樹
愛知県立三好高等学校	はな い 花井	かずし 和志

平成6年度より施行される高等学校の学習指導要領には、コンピュータの利用ということがうたわれている。現在は、どのようにコンピュータを利用した授業を行えばよいのか、その暗中模索の段階といえる。私達のグループを含め、いろいろな学校・研究会・グループ・個人が、教育現場におけるコンピュータの利用を研究しておられることと思う。

本稿では、我々のグループを例にとり、ソフトウェアの開発の仕方を論じる。

コンピュータにはいろいろな側面があるが、今回は、プログラムの作成および使用に話を限定する。

新指導要領では、数学科でコンピュータを教える内容としては

1. コンピュータの簡単なアルゴリズム
2. コンピュータによるデータ整理
3. コンピュータによるグラフィック処理

といったものとなっている。

昔、いわれていたような、「コンピュータを使用するならば、そのプログラムを作らなければならない」の言葉は、「ワープロソフト」、「表計算ソフト」、「データベースソフト」などの有用ないわゆるアプリケーションソフトが市場に出たことによって、個人のレベルでは、通用しなくなった。このため、ビジネス用のコンピュータが急速に普及した。

しかし、「コンピュータ、ソフトなければただの箱」の言葉は、常に生きており、コンピュータの普及のためには（一太郎のような）優秀なソフトの存在が必要不可欠であるといえることができる。

誰かが、ソフトウェアを作成しなければ、コンピュータに考えている仕事をやらせることはできない。いかにコンピュータ利用という掛け声を掛けても、

普及、実施に対しては、ソフトウェアの有無を無視することはできない。

現在、教育界にも、優秀なソフトウェアの登場が望まれている。

ここで、大きく問題となるのは、先生方一人一人の教え方の違い、授業を受ける生徒一人一人のレベルの違いである。この組合せは、数万通りに及ぶかと思うが、これに対し、たった1種類のソフトウェアを使ってすべて教えることができるかどうか。また、1種類（ある意味での画一化）にできるかどうか、かなりの論議が必要であると思う。

コンピュータメーカー各社が、学校向けのコンピュータシステムに、オーサリング（教材の構成を自分でやる）システムを組み込んでいるのは、この事情を意識してのことだと思う。

それでは我々のグループが実験をしている教育用ソフトウェア開発は、どのような組織・手順によって行われ、結果がどのようなであったかについて述べたいと思う。

本グループは、平成2年度に発足した。愛知県立高等学校数学科教員の有志と、コンピュータ専門学校との提携によって、グループが構成されており、教育現場で使える数学教育用のソフトウェアの開発（実験）を目的としている。

グループの分担は、教員側が、ソフトウェアの大まかな内容（使用分野・画面の表示・入力の設定など）と評価を担当し、専門学校の学生が、ソフトウェアの細かい内容（表示の仕方など）と作成を担当して、相互に連絡を取りながら、ソフトの作成・評価、改善を行っている。

このような形態での活動は、教育委員会、教科書会社などが、ソフトウェア開発業者に教育用のソフトウェアを発注する際のモデルケースとして参考になると思う。

また、ソフトウェアの開発には、技術力の蓄積という要素が無視できない（昔のゲームソフトはただ玉が飛んでブロックを消すだけのものであったが、今は背景の絵など細かいところまで凝った表示がされている。ソフトウェア技術が、前のソフトの一部を利用して別のソフトを作ることができるようなものであり、ソフトウェア会社にその技術の蓄積がなされているからである）。技術の蓄積は、早く始めるほど多く蓄積することができる。より早い時期に始めるということだけで大いに意味がある。更に専門学校がもっている他分野での技術も流用できる。

開発の詳細についてはターゲットマシンを NEC の 98 シリーズ（VM以降）とし、開発言語をCとしている。

98 シリーズを使用する理由としては、やはり最大のシェアをもっている点である。ある一部の学校でしか使えないソフトウェアを作成するよりは、より多くの学校で使うことができるものを作成しようということが、狙いである。

C言語を使用する理由としては、技術の蓄積が容易であることや実行速度が速いなどの点をあげることができるが、BASIC等の言語を使用する可能性も考慮にいれている。

さて、これより昨年度の開発の概要を説明する。

昨年度は、初年度ということもあり、教員側・専門学校側からそれぞれ6人程度の希望者を募り、手探りで開発を始めた。

プログラムを開発する大枠として、なるべくコンピュータでできないことを取り上げるとともに、各校の状況（コンピュータの普及等）を考慮して一斉提示できるものということになった。

まず、各教員がどの分野でどのようなソフトウェアがほしいかということ、それぞれまとめて、全員の前に提示した。

これを、その目的や有用性・アイデアや困難な面について全員で検討し、初年度であったため、全提案を受け入れ、それに興味を感じた専門学校生と教員がチームを作りソフトの開発をした。

内容は

1. 2次関数のグラフに対する
トレーニングプログラム
2. 2次曲線（だ円）に対する
トレーニングプログラム
3. 1次変換の固有値、固有ベクトルの
講義補助プログラム
4. 2項分布と正規分布の関係を
グラフより把握するプログラム
5. 空間内の図形表示プログラム
（円錐曲線の説明）

7月までにおおまかな内容を決定した仕様書（シナリオと呼んでいる・次ページの資料1）を教員側が出し、それをもとに、専門学校側がプログラムの作成。以後ほぼ2カ月に1回の割合で、ソフトウェアの中間発表・全員による評価の繰り返しによって、12月までにおおまかな完成（業界ではβバージョンと呼ぶ）を成し遂げた。

この結果、以下のようなことがわかった。

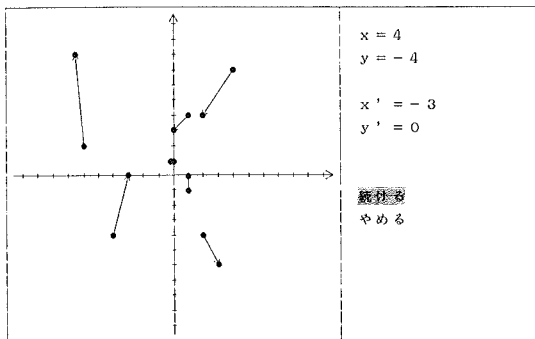
1. 期間はほぼ半年で1プログラムの作成が可能であること。
2. シナリオ作成者は、シナリオを書けばほぼ仕事の80%が終了すること（複数のプログラマと一人のシナリオ作成者が組むことによってよいプログラムを量産することも可能である）。
3. あらかじめ、対象分野の限定・教授方法の検討・効果の予測をしておかなくては、よいシナリオが書けないこと。
4. シナリオの作成は、自分の授業に使用することだけを考えて一般性を考慮にいれなければ、授業の日案および板書計画を書く程度の作業であるが、一般性などを考慮に入れるとむずかしくなること。
5. プログラムの作成に対し、専門分野の知識については、プログラマ側にきちんと伝達しなくてはいけないこと。
6. プログラムを使用するにあたっては、最初にその内容および使用法をきちんと説明すべきであること。
7. したがって、各プログラムの操作の仕方が違くと本来の授業の内容に入る前の段階で時間をとられるため、操作法の統一をはかるべきであること。
8. プログラム使用者側ではプログラムの修正を容易に行えないので、メンテナンスの体制を充実さ

せるべきこと。

この中で、もっとも素晴らしい成果といえるのは、仕事を分担することによって、プログラムをまったく知らない教員でも、自分の欲しいプログラムを手に入れることができることがわかったことである。

資料1

タイトル：固有値固有ベクトルを使った1次変換



画面の説明

行列 $\begin{pmatrix} 1 & -0.25 \\ -1 & 1 \end{pmatrix}$ によって点が変換される。
右側のテキストエリアに入力。
下に変換された点の座標がでる。

教師の説明の要点・留意事項

はじめてこのシステムを使うわけなので、全員にシステムの内容を理解させることを主眼とする

生徒の作業

点の座標を計算すると確かにコンピューターが示す点になることを計算によって確かめる。

作成上の留意点

点は好きなだけ入力できるように。
このルーチンを抜けるための操作をどのようにするか検討を要する。(特殊な値を代入するとルーチンを抜けるというソフトも多いが、生徒に何をやっているか理解させる上で、適当でないように思う。

1例として続ける・やめるをキーで選択する方式をつけた。

続けるの所で点滅
リターンを押すとカーソルがxの所に移動
リターンを押さなくて直接数字を入れたとしても同じ
↓キーを押すとやめるの所で点滅リターンを押すと次のルーチンへ移動する。

CAI ソフトの操作性について

I. 操作性の重要性

近年のコンピュータソフトの開発に関して、バージョンアップのサイクルが大変短くなってきているように思われる。これも様々な必要性からくるものであるが、そのプログラムソフトの操作性をみると、そのソフトの初期のメニュー画面のレイアウトや、操作性の全体の流れは、5～6年してもあまり変わることがない。そうしてみるとハードウェアに関してはプログラムソフトの互換性を重要視しようとする点と、ソフトウェアの操作性はバージョンアップしても基本的な流れは堅持しようとする点とは、大きな共通した柱になっていると思えてならない。ハードウェアではソフト財産がネックとなり、新しいチップで互換性を考慮するか、もしくは並列(切り換えて使用)で使用している。これもユーザーサイドにたった発想であろう。昔、N社のMS-DOSをベースにしたマシンが発売されたが1シリーズで消えてしまった。そのとき、ユーザーではBASIC用ソフト資産が多く、MS-DOS用ソフトの少なさやMS-DOSの実感といったものがユーザー側になかったように思われる。すなわち、マシン開発の動向は開発会社とユーザーの接点の中に存在することになる。他方、ソフトの面に関して、バージョンアップが行われる度にユーザーが心配することは、バージョンアップしたことでどれだけのメリットがあるかということである。あるJ社のワープロソフトで、速度の点やディスクの入れ替えのわずらわしさのために新バージョンを使用せず、旧バージョンを使用しているユーザーを見たことがある。新バージョンに大金を使ってまで利用しようという価値観が見いだせなかったのだろう。また、新バージョンにそれなりの価値観が存在したとき、次にくる問題として操作性がある。これはどのソフトも配慮していることであり、ユーザーから大きな問題として取り上げてはいないが、操作性の大幅な変更があった場合にはユーザーにとっては大きな問題となろう。漢字変換システム(FEP)では様々なシステムが考えられるが、一度そのシステムに慣れてしまうと、他のFEPは使用したくないと思うのも操作性が大きなウェートを占めている一例であろう。よって、CAI教育ソフトを開発するに当たっては、メニューやカーソルの動き、全体のレイアウトなど現在使用して

いるソフト等を参考にし、人間が特に直感的に感じる感覚を重要視し、開発にあたらなくてはならない。

II. メニュー方式について

(1) なぜそこにメニューが存在するのか？

教育ソフトの開発にあたっては単純なトップダウン的なソフトから構造化を意識した Tree 構造のものまで様々である。複雑な構造になればなるほどブロックごとのモジュール化を考えなくてはいけなくなるであろう。ソフト開発者にプログラム作成を依頼する場合にも、絵コンテを作りコースウェアを考えなくてはいけない。そんなとき、必ず出てくるのが「選択」である。この「選択」が分岐点となり構成される。1つの迷路を考えたとき、(START)→(END)までの分岐点が0のものならば、あらゆるコースの構築は不可能となり、付加価値をつけるには極めて希薄な存在となろう。もっとも、こういうものは迷路とはいわないが……。

(2) メニューの流れとしての種類

メニューの中には、Yes, No だけの単純なものから、メニューブロックを組み合わせた Tree 構造のものまでいろいろ考えられる。ここで考えなくてはいけないのは、メニューは選択でもあり、入力も要求するものでもあるということである。入力機器によっても入力概念が異なってくる。マウスからの入力の場合、キーボードで入力の場合、他にもディスプレイ、バーコード等様々な「入力」がある。構造としては Yes, No の二者択一のものから多種選択するものまでである。この中で大きな流れを2つに分けると1つはソフト側で強制的に流れを作っていくタイプと、もう1つは自由に選択できるタイプがある。自由選択の中には HELP のような強制割り込みのようなものがある。

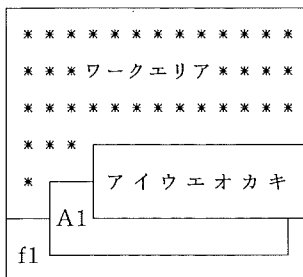
(3) 画面・視覚としてのメニュー

強制的選択	自由選択 (割り込み処理)
1……立体図形 2……二項分布 3……固有ベクトル 4……2次関数 5……2次曲線 6……終了 1から6まで選択	A B C D E F G H I J f1 f2 f3 f4 f5 f6 f7

表1 上記メニューのみ
表2 どこからでもメニュー選択

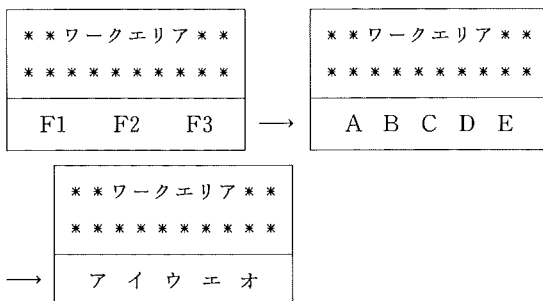
目的によって表1,表2のようになるが、現在表2が主に中心となろう。メニューのゾーンが重ねあり、かなしかに違って異なってくる。ここではワークエリアを残し、メニューをどのように変化させるか説明すると

重ねありの例



重ねる場合、メニューの変化した足跡がわかるように Tree 構造の根元の部分を表示するのが特長。

重ねなしの例

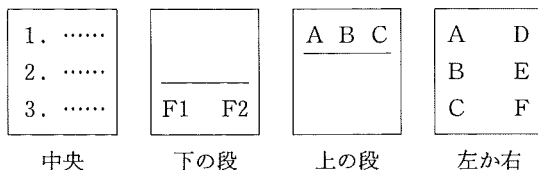


特殊キー (CTRL キー, SHIFT キー) の組み合わせにより変化するブロックごとに色などをつけてメニューの位置を表現するものもある。また、選択していくと同一位置 (下の段など) に、それぞれのメニューが表現される。

無視覚メニューについて

特殊キー (CTRL, SHIFT) の組み合わせによって選択するもの。これは、使用頻度が高くスピードを伴うものによく使用される。難点は組み合わせを暗記しなくてはならないことである。

(4) メニューの位置



III. 習慣性をどのように位置づけ利用するか

CAI ソフトを動かしたとき、ユーザーが日常どのような視覚体験をしているかによってもメニュー体型が異なってくる。ビギナー（初心者）であれば無視覚メニューなど使えるものではない。また、かなりの経験者になると日常使用しているソフトの操作が身につけているため、メニュー形式が変わると混乱を招く場合がある。特に辞書などの使用では、だいたい下の段を使用しているが、操作方法が異なるため使いにくさがでてくる。まとめてみると
まったくの初心者 ……日常の本の目次等の使い方
ワープロ専用機使用 ……メニュー、辞書変換等
パソコンを使用 ……よく使用されるソフト
などに注意し、組み立てていくのも1つの方法であろう。ここでキーの選択を少なくする方法としては、よく使用するキー、例えば、「リターン」、「ESC」、「↑ ↓ → ← の矢印キー」の役割を常に決めることで習慣化することができ、より少ないキー選択で実行することができる。また、メニューに出てくるパターンを一定化することでも習慣化することができる。

IV. 情報と自由度

メニューの中で、自由度のあるものと自由度のないものとあったが、情報としてどのような効果があるか。

自由度が高い…長所

選択項目が多いことにより、より多くの組み合わせが可能となる。

短所

組み合わせが多く情報量が多いため使い込むには初心者には困難。

自由度が低い…長所

短時間で操作を覚え、コンピュータの方で導いてくれるので、操作知識は少なくてすむ。

短所

目的の選択まで最短距離をとりにくい。使う側にとって組み合わせが伴う創作活動は不向き。

以上より、学習としては自由度は低い方がよく、ツールとしての使用では自由度が高い方がよい。このどちらも操作性について配慮しなくてはならない。

CAI ソフトを共同製作して

I. 制作の過程において

私は、2次曲線（だ円・双曲線・放物線）の導入を取り上げることにした。2次曲線は、定着が悪い分野の1つである。それは、性質を探るにも代数的な面からのアプローチが多く、円や直線のように図形との関係が頭にイメージされてこないからであろう。そこで CAI を利用し、図形的な面からのアプローチをふやしながらか少しでも改善できないかと思ったのである。とはいえ具体的なアイデアはなく、「とっつきをよくするために、導入部分で定義に従い作図させ、それをうまく評価するようなもの」といったくらいの漠然としたものだった。それに対し開発側の K さんがおもしろいアイデアを出してくれた。画面上の定点（だ円の焦点）とマウスのカーソルとの距離は、簡単に表示できるというのだ。コンピュータにそれほど詳しくない私は少々驚かされたと同時に、予想もしないことがいろいろできそうで、これからが楽しみになってきた。そして、それを利用してシナリオを作ってみることにした。

第3回目、私は20分ほどのシナリオを考えてきた。内容は定義に従ってだ円を作図させた後、円との関係を考えさせるもので、次の点を考慮した。

- ・目的をはっきりさせる。
- ・試行錯誤をさせる（ヒントも与える）。
- ・視覚的なイメージを大切にする。
- ・説明や質問を与えながら進めることができるようにする（一斉提示にも利用できるため）。

シナリオは絵コンテ風にしてプログラマの K さんにもなるべくわかりやすいものにした。

さて、実際に制作にはいるといろいろな問題点が出てきた。例えば、作業内容の指示をどうするか。補助教材を利用した授業の場合、最初に十分説明して内容を理解させることができるかどうかポイントとなる。画面上に並ぶ文字が40字を越えると生徒は読まなくなるという報告を耳にしたこともあるが、確かに画面は見にくくなる。プリントや黒板を利用する方法もあるが、今回はいろいろ考えた末に、最小限の指示は画面上にも示すことにした。キーの割り振りやメニューなどの操作性についても配慮せねばならず、試行錯誤を繰り返した。またプログラムを作る側にも多くの苦労があった。例えば、図4（STAGE 2）で点 P と F、F' を結ぶ線分が生徒の

描いた軌跡をワイパーのように消してしまうのである。私にとってみれば何ということはないが、実行速度やプログラム自体のスマートさということもあり苦勞させられたようだ。そして何より一番泣かされたのは処理速度の問題である。数学用のCAIソフトである限り、数値計算は避けて通れない。最近の16ビットパソコンがいかにか速くなったとはいえ、実際の計算を反復しながらグラフをアニメ的に見せるのには限界がある。数値演算プロセッサがあれば速度の面では開発も楽になり、プログラムの全体的な完成度も高められるであろう。現場の教師が自分なりに手を加え、ソフトをカスタマイズする場合も考えると、やはり基本性能に余裕のあるハードを標準としたいところだ。というわけで、このように私の知らない面でも多くの苦勞があったのだが、それをものすごい集中力（何時間でも没頭してしまう）と様々な工夫でクリアし、何もなかったように見せてしまうところはさすが本職であると感じさせられた。

今回5, 6回の全体会があり、そのつど、状況報告や改良点等の討議がされた。開発を進めると、つい制作者らの一人よがりになりがちであり、互いの意見交換が必要となる。そういった意味で、より綿密に会をもち討議ができればよかったように思う。

CAIソフト（だ円の導入）

<目標>

だ円の幾何学的な定義とその概形を把握する。またいくつかの簡単な性質について理解する。

<学習過程>

STAGE 1 定義に従ってだ円の概形を描かせる。

HINT 各焦点からの距離を代入させ、それを満たす点を求める。

STAGE 2 だ円の正しい軌跡を示す。

STAGE 3 だ円の定義を説明する。

STAGE 4 距離の和を一定に保ったまま焦点を移動させた場合のだ円の変化を考える。

STAGE 5 円とだ円との関係を考える。

(縮小・拡大)

(画面の例)

図1 STAGE 1

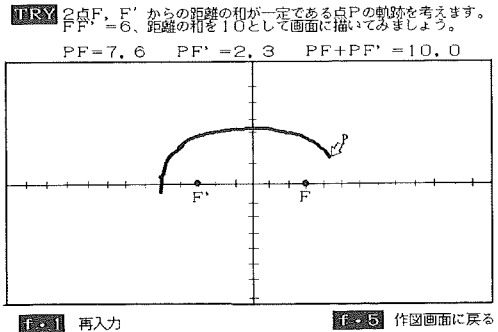


図2 HINT (値を代入)

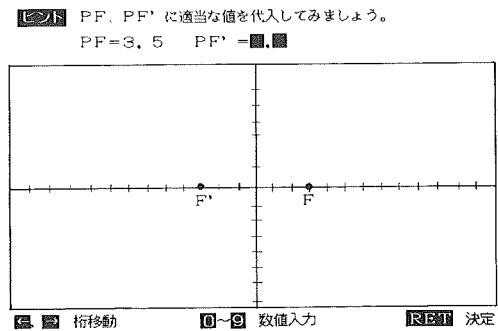


図3 HINT (作図)

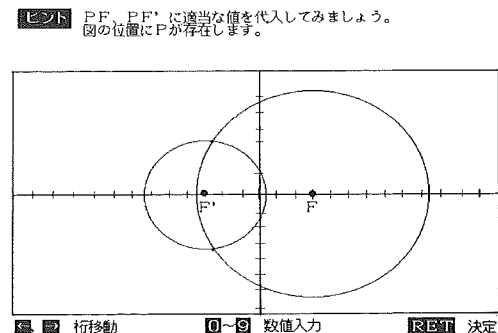
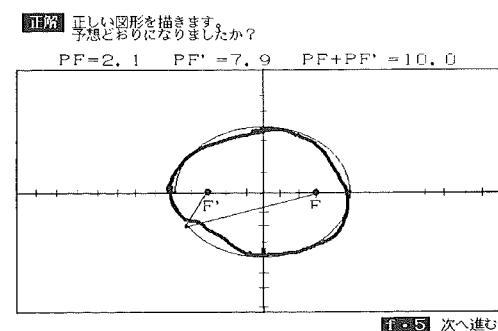


図4 STAGE 2



II. 制作を終えて

制作を通して感じたことをいくつか挙げてみたい。コンピュータの詳しい知識なしで、自分が欲しいと思ったもの・生徒の質や時流にあったものができ、自分の流儀ですぐ使えるところが一番うれしい。つまり、技術主導型（こういうことがコンピュータでできるから…）で制作を進めるのではなく、現場の求めているもの（こういうことをコンピュータでできないか…）を追求する形で進めていけるところがポイントである。そのために、その場面を通して何を考えさせたいという教師の意図をプログラマが把握することが重要となる。教師の立場だからこそ授業の過程や生徒の動向がわかるのであり、一般の人には想像しがたいところもある。実際、ソフトを利用した授業の後でのアンケートと私自身の反省をSさんに見せたところ、生徒の反応や教師が授業のたびに指導法を改善していくところなどは、開発側には見えない部分であり、大変興味深いとのことだった。教師は自分なりの教え方のノウハウをもっているだろうし、生徒達にもここがもっとこうなればといった欲求もあるだろうし、そうした様々な立場の人達の意見交換場として、この会は有意義だったと思う。また、一口にコンピュータの技術者といっても、通信から映像・ロボットその他様々な分野があり、CAIの開発も単に教師とプログラマではなく、そういった様々な分野のノウハウを集約していくことが望ましいだろう。

III. 授業を通して

実際に授業を行うと、次のようなことがわかった。PC 9801 が24台とPC-SEMIが備わった簿記計算室で行ったのだが、意外にキー操作でもたつく生徒は少なかった。また、特に女子生徒達が夢中になって取り組んでいるのは意外だった。また、習熟度や集中力の低い生徒も、このような作業型のソフトには熱心に取り組むようだ。ただマウスの扱いは難しいようで、なかなか滑らかな曲線は描けなかった。珍しさもあり画面に落書きをする生徒や先にどんどん進んでしまう生徒も何人かおり、そういう点でキー操作のいくつかは暗号とした方がよさそうだ。最初に十分な説明を与えれば生徒は一人で学習できるが、途中からは画面をこちらでロックして、皆で授業を進めた方が効果は高いようだ。

今回は、コンピュータをブラックボックスとして利用した。このことに関して、私自身にも多少抵抗があったが、子供達は予想以上に馴染むのが早い。今の電卓くらいの感覚で認識されていくのだろう。もしこだわりが生まれたら、そこで数学的に解説をしてやればよい。もしかすると、最初から数式ばかり扱うよりもより教育的といえるのかもしれない。

最後に、今回の制作を通して、貴重な経験をもつことができた。協力をしてくださった専門学校の方々やKさんに心から感謝するしだいである。

