

多人数初心者向きプログラミング教育システム の基本設計について†

米 沢 宣 義^{††} 志 村 武^{††} 南 敏^{††}

本論の最終目標は、多人数初心者に対するプログラミング教育のCAI化にある。ここでは、CAIの開発に先だってCAIの基本構成を検討するため、実際に人間教師による2種の授業方式と教育支援システムによる比較実験を行った。実験から、各学習者集団の学習特性の違いを明らかにし、これを基に十分な教育効果を保障するCAIの基本構成を提案する。

1. ま え が き

情報処理教育の重要性が高まるにつれて、各大学におけるプログラミング教育の受講者数は年々増加の一途をたどり、その教育体制も「多人数多クラス」化の傾向にある。この「多人数」化は、教師の負担の増加と教育効果の減少を伴い、「多クラス」化は、担当教師の個性と教授法の違いによってクラス間の学習状態に格差を生じる。最近パソコンの普及も手伝って学生のプログラミングの「知識の多様性」も加わり、従来の講義演習による一斉授業方式では思ったとおりの教育効果を望めないのが現状である。さらに、一斉授業方式が成り立たなくなるのは時間の問題と思われる。そのため、「多人数」「知識の多様性」の環境にあっても教師の負担を軽減し、かつ十分な教育効果を期待でき、「多クラス」間にも平等な教育指導が可能な教育支援システムが必要となる。

実際、各方面で教育支援システムの研究開発¹⁾が行われており、その一つに教育対象となる言語のエラー分析から言語の特徴を明確にし、これを基に教育支援システムを開発する試みがある²⁾。しかし、限られた時間内に初心者と言語の修得からプログラミング応用まで教育する場合、プログラムを数多く作成していく実践式とプログラミングの時間を多少犠牲にしても文法知識の定着および机上演習に重みを置いた方式のどちらが最終的にプログラミング能力が身につくかという検討はなされていない。

一方、この二方式の優劣を判断するためには、学習

者集団の学習状態を定量的に扱う手法が必要となる。しかし、学習者集団中の個々の学習者を評価する手法はあっても³⁾、集団の学習状態の推移や集団間の特性の違いを表現する簡便な手法はない。

ここでは、これらの点に着目して両方式を実際の授業に適用し、その各学習者集団の学習状態を定量的に表現することによって、言語修得とプログラミングに必要な時間の配分および授業方式はどうあるべきかを検討した。その結果、プログラム作成に関する応用力が身に付き、さらに文法エラーと実行時エラーの発生率が小さく、エラーが発生しても自力で除去できるという点では、計算機を自由に使用してプログラミングを行う方式はあまり得策でないことが裏付けられた。むしろ、計算機の使用を制限した方式、すなわち文法修得と簡単なプログラムの机上演習および計算機実習の三つのモードをバランスよく短期間に集中して行う方式に有効性が認められた。さらに、この方式を教育支援システム⁴⁾⁻⁶⁾で細かく制御すれば人間教師が直接授業を行わなくても人間教師と同程度あるいはそれ以上の効果を得ることがわかった。

なお、最後にプログラマ適性と授業形式およびプログラム作成能力の関連について簡単に述べる。

2. 教育支援システム

CAIの基本構成を検討するため人間教師による授業方式と教育支援システムによる方式の比較を行ったが、このときに使用した教育支援システム‘KUCAS’の概略を説明する。図1にKUCASのモデルを示す。KUCASは、ドリル & プラクティスとプロブレムソルビングの二つのモードからなる。

ドリル & プラクティスモードは複数のブロックからなり、各ブロックはメインとサブのブロックに分かれている。言語修得とプログラムの組み立て方法はこ

† Basic Plan of Instruction System on Programming Course for a Large Number of Beginners by NOBUYOSHI YONEZAWA, TAKESHI SHIMURA and TOSHI MINAMI (Department of Electronic Engineering, Faculty of Engineering, Kogakuin University).

†† 工学院大学工学部電子工学科

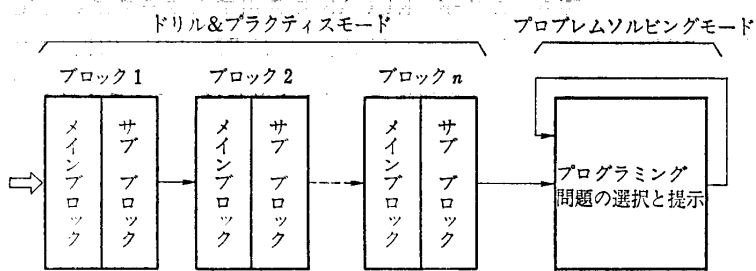
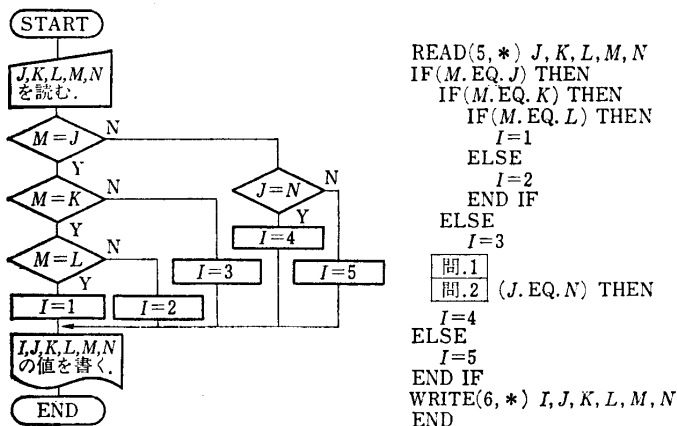


図 1 KUCAS のモデル
Fig. 1 Model of KUCAS.

下記のフローチャートに対応するプログラムを完成せよ。



例
Example.

のモードで行う。メインブロックは、学習者全員に同一の問題を複数提示し、この結果を評価して各学習者の苦手な部分を検出する。サブブロックでは、この苦手な部分を補うためにヒント付きの補足問題を提示する。

全ブロックを通過すると、次はプロブレムソルビングモードになる。このモードでは全ブロックの解答状況の評価し、理解していないと思われる部分を補うためにプログラミング問題を順次提示する。学習者はこの問題を実際に計算機を利用して完成する。

3. KUCAS の教案項目

KUCAS の指導テキストは、本学情報工学コース 1 学年の情報処理基礎演習（必須科目）用に構成したもので、6 個のブロックとプログラム作成問題からなる。ブロックは一貫して、プログラムのなかで使われている命令の生きた文法を学ぶとともに個々の命令のつながりを体系的に把握できるように小さな基本プログラムを中心に構成した<例>。

演習は半期で、授業回数は 1 回を 90 分として 12~13 回ある。最初の 2~3 回の授業で教師が FORTRAN の概略を説明するが、その後は受講生の自主性にまかせ、参考書を調べさせながらテキストを解かせる。個人によって学習進度は異なるが平均 4 回で 6 ブロックを終了し、あとはプログラム作成を行うように編集した。

テキストは次のような構成になっている。

ブロック 1 【算術式と基本入出力】（メインブロック設問 28 個）

- 内容 1) 実数型と整数型の定数と変数
2) 型混合の算術式と代入文
3) 並びによる入出力
4) FORMAT 付き入出力

ブロック 2 【制御文】（メインブロック設問 19 個）

- 内容 1) GO TO 文
2) ブロック IF 文
3) 算術 IF 文、論理 IF 文
4) DO 文

ブロック 3 【配列】（メインブロック設問 20 個）

- 内容 1) 入出力と配列（1, 2, 3 次元）
2) 配列を用いた基本問題

ブロック 4 【文字処理】（メインブロック設問 19 個）

- 内容 1) CHARACTER, DATA 文
2) 文字の入出力
3) 文字連結

ブロック 5 【副プログラム】（メインブロック設問 13 個）

- 内容 1) COMMON 文
2) 文関数
3) 関数副プログラム
4) サブルーチン副プログラム

ブロック 6 【フローチャート】（メインブロック設問 19 個）

- 内容 1) プログラムの組み立て問題

4. 教育効果の定量的表現

測定の対象となる学習者集団の学習状態の違いを明

確に表現するため、次のデータを利用した。

(1) 学習者個人ごとのプログラム作成数と CPU 使用時間

(2) 2 度の一斉試験によるテスト得点

(3) プログラミングエラー

(4) 学習者に対するアンケートと適性

(1) から一つのプログラムを完成するために使用した平均 CPU 時間を求め、これとテスト得点の関係を導く。

(2) から CS-T 平面図を作成する。CS-T 平面図とは、注意係数 CS とテスト得点 T を軸にした平面図上に各学習者の位置づけを行い散布図を描き、さらにこの散布図に重ねて集中楕円 (等確率楕円) を書いたもので、楕円の形状で学習者集団の状態の把握と推移を視覚的に観察するための平面図法である。

(4) のアンケートは授業開講前のプログラミング経験を調査するために行った。

5. 実験

本学情報工学コース 1 学年の情報処理基礎演習 (必須科目) は、1 クラスを約 30 名として 4 クラスに分割されている。このクラスに KUCAS を利用した授業方式と教師による授業方式を適用し、その学習効果の違いを測定した。

5.1 実験環境

測定は 57, 58, 59 年度の 3 回行ったが、57 年度は FORTRAN, 58, 59 年度は FORTRAN 77 であった。プライバシーの問題もあるので 4 クラスを A, B, C, D とし、57 年度は A 57, B 57, C 57, D 57 のように添字を付けて表す。

各クラスの受講生はランダムに均等配分されており授業開講の初期状態では、クラス間に能力差はないものとする。実際、プログラミング経験者も各クラス A 57, B 57, C 57, D 57 に対してそれぞれ 7, 8, 8, 8 人の割合となった。測定は、必須科目の授業を対象とするのでかなり信ぴょう性の高い測定値が得られる反面、授業時間外は受講生を拘束できないのでクラス同士の干渉や経験者の他への影響により測定値に歪みがある。ただ、受講生は測定対象となっていることを知らないで、干渉は最小限に留まるものと思われる。本実験では、授業時間外の経験者による他への影響やクラス同士の干渉は 4 クラス全体に平均化されているものとみなす。

クラスを担当する教師の指導能力差も測定に対する

歪みとなるが、KUCAS を適用するクラスに対しては、最初に概略を説明し、後は受講生の自主性にまかせる方式であるので教師の指導力はほとんど影響しないものとする。

なお、どの授業方式も同一の計算機システム、ACOS 650 (主記憶装置 4 MB), 磁気ディスク (317 MB×4), TSS 端末 (30 台), プリンタを使用した。

5.2 授業方式

[1] 57 年度授業方式

(I) A 57 は実習中心の授業方式で教師 a が担当した。教師 a は、5 回の一斉授業で基本文法を教え、残り 8 回と補講 3 回の計 11 回で BATCH と TSS を駆使したプログラミング実習を実施した。

(II) B 57, C 57, D 57 は、教師 b が最初の 3 回の一斉授業で文法を講義した。

(III) その後、B 57 は教師 c が授業を引き継ぎ、続く 4 回の一斉授業で KUCAS 専用テキストの演習問題を解かせ、残り 6 回をプログラミング実習に割り当てた。プログラミング問題は、教師 c が KUCAS 内の問題群から適当に複数個選び全員に同一の問題を提示した。

(IV) C 57, D 57 は教師の介入を受けず、KUCAS の指導だけで学習を行った。なお実習は B 57, C 57, D 57 とともに BATCH 処理を用いた。

[2] 58 年度授業方式 (各クラスをそれぞれ A 58, B 58, C 58, D 58 と略記)

(I) A 58 は教師 a, B 58 は教師 d が担当した。授業は A 57 と同一の方式を採用したが、実習については A 58 で 5 回、B 58 で 1 回分の補講があった。さらに A 58 は授業時間外の計算機利用を認めていた。

(II) C 58, D 58 は C 57, D 57 とまったく同一の授業方式を採用した。

5.3 試験方法

[1] 57 年度一斉試験

上記 4 クラスに対して 7 回目と 13 回目の授業終了後に一斉テストを実施した。以降これを 1 回目テスト、2 回目テストと呼ぶ。1 回目テストは基礎問題を中心に出题し、2 回目テストは基礎問題と応用問題に分け、基礎問題は 1 回目テストの類題で構成した。ただし、2 回目テストの基礎問題は、KUCAS システムにかかわりのない教師 a が作成し、また応用問題は難易度に一般性を持たせるため第 2 種情報処理技術者試験問題集から抜粋した問題を含ませた。

[2] 58 年度一斉試験

表 1 57年度試験結果 (経験者を除く)

Table 1 Result of test in 1982 (except person experienced in programming).

(a) 1回目 (a) The first test				(b) 2回目 (基礎) (b) The second test on basis				(c) 2回目 (応用) (c) The second test on application			
クラス	人数	平均	偏差	クラス	人数	平均	偏差	クラス	人数	平均	偏差
A57	17	41.77	12.16	A57	17	64.27	12.89	A57	17	17.65	15.16
B57	22	63.64	14.97	B57	22	70.02	9.10	B57	22	27.27	25.97
C57	18	59.07	14.04	C57	18	70.16	7.69	C57	18	36.67	34.07
D57	13	73.33	13.64	D57	13	76.64	8.76	D57	13	50.77	23.03

表 2 58年度試験結果
(経験者を除く)Table 2 Result of test in 1983
(except person experienced in programming).

クラス	人数	平均	偏差
A58	13	53.56	19.90
B58	19	55.55	20.24
C58	14	52.65	19.76
D58	19	60.62	19.90

13回目の授業終了後、全クラスに1度だけ実施した。問題は、公平を期するため演習の授業とはまったく無関係の教師 x が出題した。なお、両年度の試験問題は○×式の客観テストを避け記述式を採用したので、客観的に採点しやすくするため1文字でも間違えば誤りと判断した。例えば、プログラム作成問題では、学生が書いたコーディングをそのまま計算機に入力したとき正しい実行結果を出せるものだけを正答とした。

6. 試験結果

第2回目基礎問題において、経験者を除いたクラス全体 (70人) の得点平均 $\bar{x}$ と偏差 σ は、 $\bar{x}=69.9$, $\sigma=10.56$ で、経験者 (31人) の平均と偏差は、 $\bar{x}=79.09$, $\sigma=8.7$ である。また、A57の経験者 (7人) については、 $\bar{x}=76.19$, $\sigma=9.04$, D57の経験者 (8人) については、 $\bar{x}=80.6$, $\sigma=10.61$ である。これは明らかに経験者のほとんどが授業方式に関係なく自主的に学習ができる能力があったことを示している。そこで

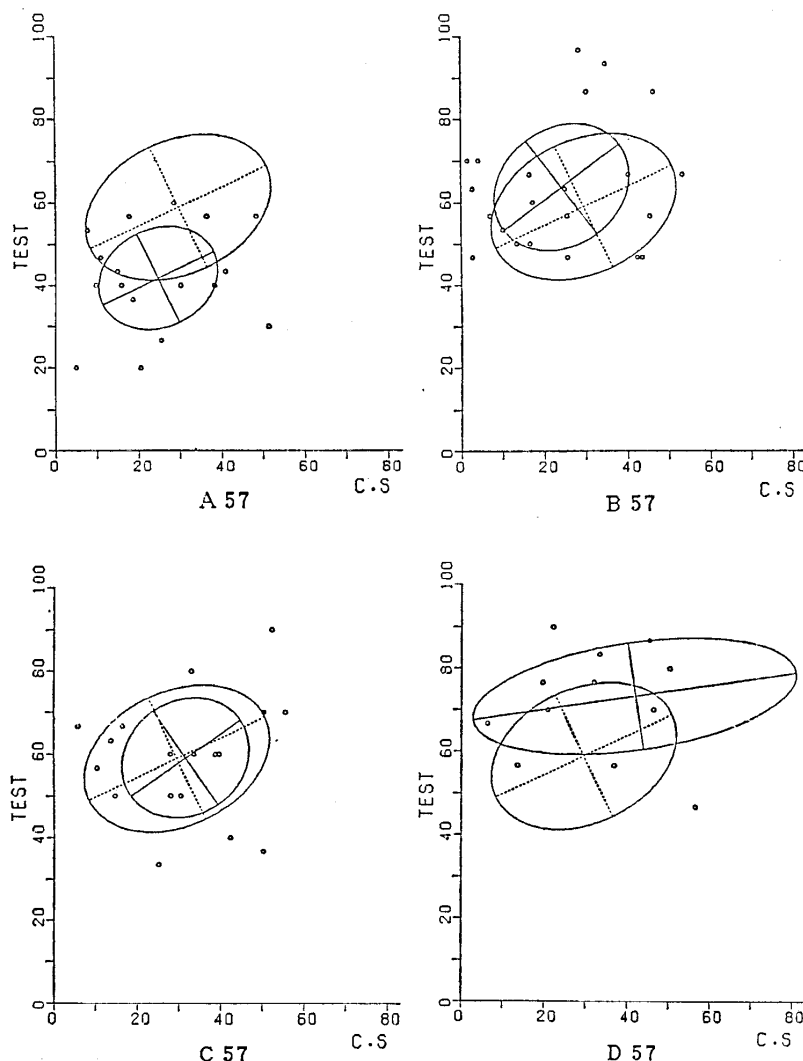


図 2 1回目テスト CS-T 平面図
Fig. 2 'CS-T' figure for 1st time test.

授業方式の優劣を決めるための分析においては経験者は除いて検討することにする。表 1, 2 は、経験者を除いた場合の一斉試験の結果である。

図 2 は 57 年度 1 回目試験結果から求める CS-T 平面図である。図中の長、短軸が破線で示された楕円は

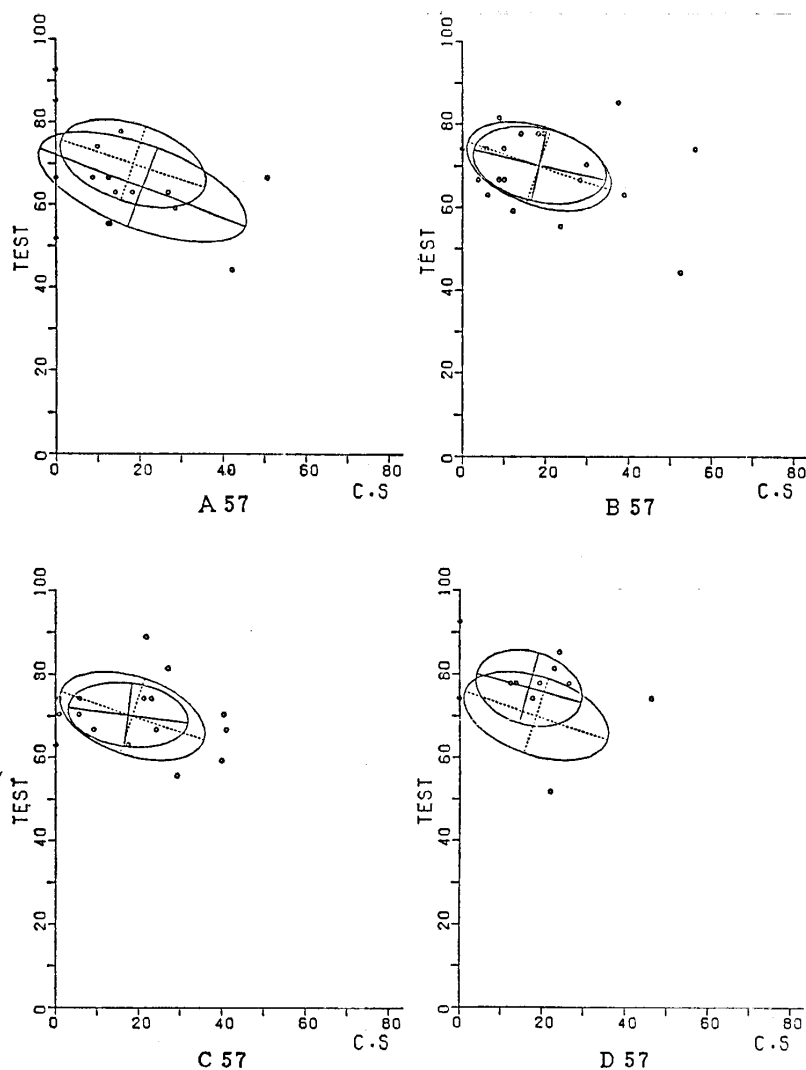


図3 2回目基礎テスト CS-T 平面図
Fig. 3 'CS-T' figure for 2nd time basic test.

表3 プログラム問題正答者数
Table 3 Number of students who give a correct answer for programming problem.

クラス	問 1	問 2
A57	6人 (35%)	2人 (12%)
B57	5人 (23%)	7人 (32%)
C57	11人 (61%)	5人 (28%)
D57	8人 (61%)	6人 (46%)

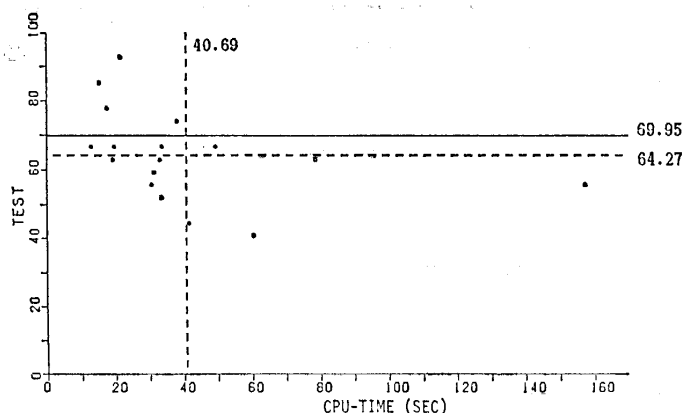
4クラス全体から求まる等確率楕円で、比較基準のため記入した。図から A57 は他3クラスと比較して基準楕円よりもかなり下方で集中し、得点も他の3クラスより劣っていることがわかる。実際、得点差の F -検定を行ってみると 5%水準で有意であった。この結果は A57 以外で行われたテキストを用いた授業方式

の有効性を示す。

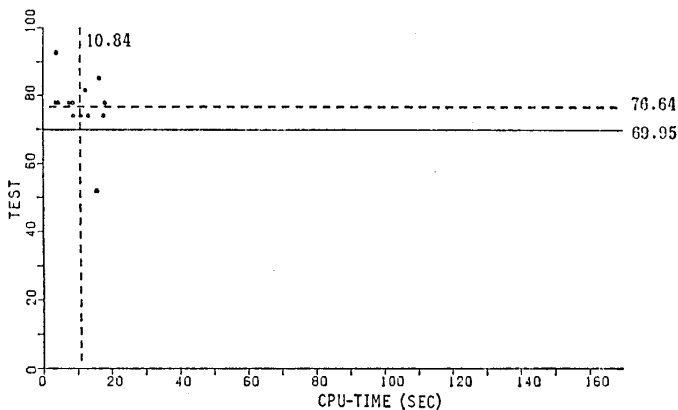
図3は2回目基礎テスト結果から求まる CS-T 平面図である。この試験は1回目試験の類題で構成してあるので図2と図3の比較が可能である。図2から図3への楕円の推移は、すべてのクラスで基礎学力の向上があったことを示すが、そのなかでも特に D57 が優れていることがわかる。しかし依然として A57 は全体平均より劣り、集中度も悪い。この2クラス間で2回目基礎テストの得点差の検定を行ってみると 5%水準で有意であった。結局、A57 は D57 との基礎学力の差をプログラミング実習で埋め合わせるまでには至らなかったのである。応用テスト(表1(c))については、得点上位2クラスがともに KUCAS を用いたクラスであった。このテストにおいて KUCAS を用いたクラスと他の2クラスとの間の得点差の検定を行うと 5%水準で有意であり、応用力を養うという面でも KUCAS による学習形式は他の授業方式よりも優れていることが明らかになった。この点については応用力が基礎学力の上に成り立つものと考えれば当然の結果と思われる。

次に、2回目応用問題中にあるプログラム作成問題の正答者数の比較からプログラム作成能力について検討する。問題は2問あり、表3は問1、問2に正答した学生の人数、()内の数値は各クラス内でのその割合を示す。問1は KUCAS を用いた2クラスともに半数以上の者が正答し、問2も D57 が優れている。結局、プログラム作成能力を高めるという点においても KUCAS の学習方式の有効性が示された。

図4は個々の学生がプログラミング問題1問完成させるために要した CPU-TIME 使用量と2回目基礎テスト得点の関係を示した図である。ただし、B57 と C57 は D57 とほぼ同じ特性であったので省略した。A57 はプログラミング問題を効率よく解いていない



(a) A57



(b) D57

図4 プログラム問題1問完成するのに要した CPU-TIME と
テスト得点 (2回目基礎) の関係
(....., クラス平均; —, 全体平均)

Fig. 4 Relation between the score for 2nd time basic test and
cpu-time spend for solving a programming problem.
(....., average score for a class.)
(—, average score for 4 classes.)

学生に限って得点も低いという傾向が見られる。これより、逆に計算機を自由に使用して問題を多く解かせることで学習効果を高めるという実践的手法は学習効率という面からも有効な手法でないことが裏づけられた。

表2は58年度一斉試験の結果である。表2はKUCASを用いたクラスD58が4クラス中最もよく、C58は最低であることを示す。しかし検定を行ってみると各クラス間に有意差は認められず、CS-T平面図を作成してみても各クラスの特徴を見いだすことはできなかった。

これは次の点に起因していると思われる。

(1) A58のプログラミング実習時間は補講を合わせるとD58の2倍以上で、さらに授業時間外での

計算機利用を認めていた。

(2) あらかじめ教授項目を定め4クラスに対してこれをすべて教授するはずであった。しかし、教師d (B58)は独自の判断で教授項目の半分程度を時間をかけて教育した。結果的に教授項目と試験項目が部分的に一致していたことを追記しておく。

一方、KUCASを用いたクラスは、

(1) FORTRANからFORTRAN 77に変わり教授項目がふえたためテキストを解く期間が57年度より2回分のび、プログラミング実習時間が縮まった。

(2) マシンダウンが多く、授業計画にずれが生じた。

しかし、このような状況下でも、KUCASを用いたクラスがA58およびB58と同程度の学習効果を得られたことは、KUCASによる学習方式は有効であったことを示している。

59年度はKUCASを適用した1クラスと、異なる教師が担当する2クラスに対して同様の測定を行った。KUCASのFORTRAN 77テキストが安定したためか測定結果は57年度と同様の傾向を示した。

7. 適 性

以上、授業方式の差による学習効果の比較を行った。

ここではさらに、最近重視されてきているプログラマとしての適性とプログラミング教育との関連について調べてみた。調査対象として先のA58、B58、C58、D58を選んだ。適性問題は就職情報誌⁷⁾に載っていた問題を利用した。この問題は各自でプログラマとしての適性を判断する形式であったが、これを四つのクラスに一斉試験の形で授業開講前に実施した。適性は四つの設問の総合成績から判断され、得点の高い者ほど適性を有するというものであった。また、今回の測定においても対象集団から経験者は除外した。

結果は、A58、B58で適性得点と試験得点に正の相関があり、相関の検定(スピアマンの相関係数の検定)から5%水準でB58において'相関あり'と判断された。

C58、D58はともにKUCASを用いた授業方式でありながら、C58で正、D58で負の相関を示した。

しかし検定によると両クラスとも相関が認められなかった。

ところが、A58とB58の合成集団については'相関あり'、C58とD58の合成集団には'相関なし'と判断されたことから、実践を中心とした授業方式の学習効果は適性に左右され、KUCASを用いた授業方式は適性との関連性はないことが予想される。

8. む す び

KUCASを用いてプログラミング学習を行った学生は、基礎学力、プログラム作成能力ともに優れ、基礎文法の修得と机上演習およびプログラミング実習をバランスよく行うことの重要性が示された。

CAIに特有な「単調性」は、CAIの適用時間が長ければ長いほど学習に悪影響を及ぼすと考えられる。やはり、コミュニケーションの媒体をかえて授業パターンに変化をつけることも必要と思われる。今後、この点も考慮しつつ、最初に行う文法の講義はビデオで置き換え、さらにKUCASを改良して本格的にプログラミング実習の自動化を進める予定である。

また、適性については58年度が初めての実験であったので、適性と言語教育との明確な関連性は見いだ

せなかった。これについてはさらに毎年繰り返し適性試験を行い、言語教育との関連について調べていく必要があると思われる。

参 考 文 献

- 1) 深川幸紀他：FORTRAN 様言語を使用できるCAIサポートシステム，信学論(D)，Vol. J63-D (1980)。
- 2) 下条真司他：FORTRAN 初等教育を目的としたCAIシステムの設計開発にあたっての調査，信学技法，ET 83-3 (1983)。
- 3) 佐藤隆博他：生徒の項目反応パターンの指数に関する考察，信学技法，ET 80-11 (1981)。
- 4) 米沢宣義他：56年度第4回情報処理教育シンポジウム資料，私立大学等情報処理教育連絡協議会 (1981)。
- 5) 米沢宣義他：58年度第6回情報処理教育の進め方についてのシンポジウム資料，私立大学等情報処理教育連絡協議会 (1983)。
- 6) 米沢宣義他：58年度第6回教育ソフトウェア説明会資料，私立大学等情報教育連絡協議会 (1983)。
- 7) コンピュータ・エージ社：就職ガイド'84，コンピュータ産業 (1984)。

(昭和59年2月18日受付)

(昭和60年9月19日採録)