

問題解決過程としてのプログラミング

荒川 淳三

要 旨

今日の文科系大学のプログラミング教育は、学生たちに情報処理の読み書き的基礎を与える教育の一環をなすものであるが、プログラムの仕様を定めてこれを設計製作し完成させる過程は典型的な問題解決過程であり、少し複雑な問題になると、計画的な取組がなされないかぎり成功は容易ではない。

一方大学においてPDCAに代表される仕事の進め方（動的知識）の基礎を教えることは重要であるが、日本の大学教育がこの面において弱体であることは否めない。

筆者は文科系大学におけるプログラミング教育を動的知識体得の場として位置づけることの意義を確信し、若干の試行結果をもとにこれを提案するものである。

1. 札幌大学経営学部における情報処理教育の現状と問題点

札幌大学経営学部では情報処理教育の一層の充実のために、1989年にカリキュラムの変更を行った。主な変更点は次の通りである。

- (1) それまで選択科目だった「電子計算機概論」を「情報処理概論」の科目名のもとに必修化した。
- (2) コボル言語 I, II, フォートラン言語 I, II に分かれていたプログラミング科目を、プログラミング論 I (2年次), II (3年次) に統合し、使用言語をフォートランとした。なお「経営情報論」, 「経営システム論」, 「情報管理論」など情報処理応用科目の多くは、新カリキュラムにそのまま引き継がれている。

以下に情報処理教育の基礎科目である情報処理概論とプログラミング論 I, II の現状と問題点について述べる。

1.1 情報処理概論の現状と問題点

情報処理概論は、新入生に対して情報処理に関する基礎的理解を広く与えることを目的に開始された。1989年度の講義で扱った主要項目は表-1の通りである。しかし学生たちは一般的な事項に対しては深い関心を示さず（教室がざわつき）消化吸收の度合いも不十分だったため、1990年度からは講義対象を大幅に狭め、情報処理概論を2年次以降のプログラミング論のための助走科目的性格に変え、今日に至っている（表-1）。

なお情報処理に関する一般的知識については、それぞれの専門科目において随時教えるのが適切であると考えている。

表1 情報処理概論の内容 (初期と現在の対比)

1989年度	1992年度
1. 経営と情報処理-1 (小規模システム例) 2. 経営と情報処理-2 (中規模システム例) 3. 経営と情報処理-3 (大規模システム例) 4. 経営と情報処理 ・アンソニーの組織ピラミッド ・システムの成長とノーランモデル 5. 情報処理技術の歴史的展開 6. 情報のエントロピーと2進法 7. 論理と論理回路 8. コード化 9. コンピュータの仕組み 10. アルゴリズムとフローチャート 11. プログラミング 12. エンドユーザー言語とDIY 13. 人口知能 14. 経営におけるコンピュータ利用の今後 15. 経営学とコンピュータ	1. 情報のエントロピー 2. コンピュータの原理 3. コンピュータの利用手順 4. 基本的アルゴリズムとプログラム ・メモリーの内容を入れ換える ・合計を求める ・整数同士の除算 偶奇判定 時間の換算 ユークリッドの互除法 ・最大値、最小値を求める ・大きさの順に並べ替える ・二分探索法 5. 2進法とデータ表現 6. フローチャート

表2 プログラミング科目履修者の推移

科目名	1988	1989	1990	1991	1992
フォートラン言語Ⅰ	44	78			
コボル言語Ⅰ	107	110			
フォートラン言語Ⅱ	20	9	23		
コボル言語Ⅱ	12	32	34		
プログラミング論Ⅰ			226	270	241
プログラミング論Ⅱ				30	34

1.2 プログラミング論の量的現状と問題点

表-2はプログラミング関係科目履修者数の推移を示すものである。

1990年度から実施されたプログラミング論Iでは履修者の大幅な増加が見られ、それ以前のフォートラン言語Iの履修者のほとんどがコボル言語Iも合わせて履修していたことを考えると、プログラミングの履修者は一挙に倍増したことになる。これには情報処理概論の必修化が大きく影響していると考えられ、新カリキュラムの成果がここに現れていると言える。

プログラミング論Iの履修者が大幅に増えているのに対し、プログラミング論IIの履修者は全く増えていない。プログラミング論Iを履修した学生が何故プログラミング論IIに進まないのかについては本格的な調査は行っていないが、非公式の聞き込み調査等から、その理由は次のように推察される。

- ・プログラミング論Iの理解が不十分で、プログラミング論IIに進む自信がない。
- ・プログラミング論では課題の消化に正規の時間外にかなりの時間が必要であり、単位取得的には効率が良くない。
- ・プログラミング論Iである程度の理解はできた。この程度でよしとしたい。

要約すればプログラミング論Iを十分に理解していない、またプログラミング論に対してあまり多くを期待していないというのが、学生たちの平均的考えであると思われる。

1.3 プログラミング論の質的現状と問題点

これまでの議論から、量的には大きな前進を示したプログラミング論Iに質的問題が残されていることは明らかである。具体的には、プログラミング論Iを履修する学生たちについて、次のような問題が指摘できる。

- (1) 講義の内容はそれなりに理解できるが、

具体的なプログラムを作成する場合どうしたらよいかわからないという学生が多い。原因はプログラムの基本形を把握していないためであるが、かれらが問題をモデル的にとらえる素養に欠けていることが背景にあると思われる。

- (2) 平均的學生は教科書あるいは参考書をほとんど読んでいない。かれらの多くは教科書が難しすぎることを訴えるが、かれらが知識を文章から吸収しこれを体系的に積み上げる能力に不足していることは否めない。

- (3) プログラミングの基礎的事項を理解していない學生が非常に多い。

例えばデータの宣言が何故必要なのか、配列の個々の要素はどう命名されるのか、入出力編集記述子の意味、無限繰り返し構造からの脱出方法など、極めて初歩的なことでさえ理解していない學生である。

この点に関しての問題の一端は、筆者を含め教える側にあるように思われる。今日の學生たちの多くは主体的に学習し成果を着実に積み上げていく能力を欠いており、彼らの教育においてはその習得状況を常時正確に把握し、習得状況をみながら対応を工夫していくというきめ細かな配慮が不可欠であるように思われる。

- (4) 學生たちの課題への取り組み方が非常に試行錯誤的である。課題の内容さえ理解せぬままに端末に向かってプログラムを打ち込み、コンパイルバグをつぶしながら何とか仕上げようとする。テストデータに対してそれらしい出力が得られると、検算もせずそれを提出する例も目立つ。

學生たちのこのような姿勢の背景には、彼らが問題解決に計画的に取り組む訓練をほとんど受けていないことが有るのではないと思われる。また教える側としては、プログラムの品質保証の重要性についての教育が不十分であることを反省せざるをえな

い。

以上からプログラミング教育をより効果的にするための方策を考えると、先ず第一に今日の学生たちの言語能力、モデル化能力、問題解決能力等を疑ってかかる必要がある。

J. Healy はテレビ時代がアメリカの子供たちの知的能力に与える影響について「減び行く思考力」の中で論じているが⁽¹⁾、同じことが今日の日本の学生たちにも当てはまるように思われる。高等学校における「私立文系」の存在がこの傾向に拍車をかけていることは明らかである。

今日の学生たちにおいて学ぶ姿勢が不十分であることも、プログラミング教育において大きな障害となっている。放置しておけば彼らは勉強しない。平均値的學生に納得の行く学習成果をあげさせるためには、彼らの習得状況を常時監視し、手取り足取りの指導を行うことが必要である。

プログラミングの課題に計画的に取り組むことの指導も、極めて重要である。闇雲に課題に取り組むなどとかこれをでっち上げるといふ姿勢では、十分な学習成果が挙げられなればかりでなく弊害が大きい。教える側は學生たちに計画に基づく仕事の進め方の重要性を十分に理解させるとともに、そのための具体的な方法論を与えることが必要である。

2. 動的知識教育の場としての プログラミング

プログラミングの課題に取り組むに当たってはこれを計画に基づいて行うことが極めて重要であるが、この主張を一步進めて、計画に基づく仕事の進め方を學生たちに体得させることをプログラミング教育の最重要目的に据えたらどうかというのが私の提案である。その理由の第一は計画に基づく仕事の進め方が何にもまして重要なためである。

筆者が以前勤務していた民間企業（新日本

製鉄株式会社）では、かつて膨大な人力を投入して社員の身につけるべき基本能力についての調査研究を行ったが、この結論としての最重要能力要素は ①IE的能力 ②組織化力 ③専門技術力 の3つであった。

IE的能力とは現状分析に基づく問題点の摘出から必要な解決策の実施までの仕事の進め方に関する理解であり、組織化力とは推進すべき仕事を組織公認の仕事として位置づけさせる能力、あるいは組織を動かす能力である。①、②をまとめて仕事の進め方としていくことも可能であり、これらの根底にあるのは周到な計画に基づく業務運営である。

新日本製鉄株式会社において、仕事の進め方が社員の身につけるべき最重要能力として挙げられたもう一つの理由は、それを身につけることが非常に難しいからである。このような一朝一夕には身につかない基礎的能力の素地は、学生時代にしっかりと与えておくことが効果的であると考えるのが、提案の第二の理由である。

第三の理由は、日本の大学が仕事の進め方に関する知識（動的知識）の教育において貧弱であると考えられる故である。京都大学の松本吉弘教授が（株）東芝勤務時代の著書で、「一般に大学におけるソフトウェア教育が企業の中で十分に評価されていないのは、静的知識のみがとり上げられ、それを駆使する動的知識への志向がなされていないからと考えられる」と述べておられるように⁽²⁾、動的知識教育の不足を痛感するのは筆者のみではない。

第四の理由は、文科系大学で仕事の進め方を体得させるための場を求めるとき、プログラミング教育はそのための数少ない、あるいは唯一の場と考えられるからである。

プログラミング言語という小さな部品を組み合わせて大きなプログラムを仕上げていく過程は本質的に複雑であり、一寸込み入った問題になると、そのプログラミング的解決には、周到緻密な計画が不可欠である。プログ

プログラミングの学習を通し、学生たちは計画に基づく仕事の進め方の絶大な効果を理解できる。

またプログラミング課題への取り組みは繰り返し行われる。計画に基づく仕事の進め方の基本は「PDCA」であるが、課題ごとにその取り組み方を評価し、次の課題にはより完全な計画を立てて取り組むことが可能な点においても、プログラミングの学習は仕事の進め方を学ぶ上で恰好である。

理科系の大学には実験科目があり、これを通して学生たちは緻密な計画の重要性を理解できるが、プログラミング学習は文科系にはない「実験」の役割を担い得る科目である。

第五の理由は、計画的な仕事の進め方を適用することによって、学生たちはプログラミングを十分に理解することが可能となるからである。計画に基づく方法は本質において分析的であり、大きな課題を小さく分けることにより、学生たちは主体的に思考することが可能になり、プログラミング自体をよく理解するとともに、問題解決の基本である分析能力も身につけることができる。

3. 具体的方法論と成果

3.1 方法論

計画に基づく仕事の進め方は問題解決における基本的な方法論と言えるが、問題解決のための基本能力を習得するにはこれを自分で主体的に体験するしかなく⁽³⁾、計画に基づく仕事の進め方についても、これを習得する唯一の方法は、個々人が主体的に体験することである。

したがって本稿の提案は、プログラミング課題への取り組みをPDCAを基本に展開することに尽きる。

具体的方法としては、課題への取組において先ず周到な計画書の作成を義務づけ、計画書が承認されてはじめて、コーディングに着

手することを認める。また課題完了時には計画と実績との対比に基づく実績報告書を提出させる。

計画書、実績報告書については、課題ごとに書式の印刷された用紙を配付するが、現在用いている用紙は計画書、報告書をB4版裏表にまとめたもので、記入項目を次の通り指定し、各項目記入のためのマニュアルを作成配付している。

- ①機能概要
- ②出力
- ③入力
- ④データ定義
- ⑤処理フロー（フローチャート）
- ⑥テスト・データ
- ⑦スケジュール（ガントチャート、予定と実績を記入）
- ⑧バグリスト
- ⑨実績評価
 - ・スケジュールの評価
 - ・プログラムの出来ばえ評価
 - ・課題を通じて学んだ点
 - ・次回に改善すべき点

3.2 成果

本方法論の適用は1992年度後半からで日は浅いが、その効果は絶大である。例えば二年次のゼミ生に適用した結果では、昨年2か月余をかけても最後まで到達した学生が半分に満たなかったプログラミング課題（散布図を用いての相関分析）において、今年度の学生たちは3週間で全員がこれを完了した。また実績評価においても「独力でやり通した最初の課題である」、「緻密に計画することの重要さがよく分かった」などの肯定的評価が出されている。

わずかな実績であるが、今後の成果に確信が持てる。

4. 今後の課題

プログラミング教育を通して「計画に基づく仕事の進め方」を指導する上での最大の問題は、教える側の負荷の増大である。札幌大学経営学部のプログラミング論 I の例では、一人の教員が教える学生数は百人前後であり、彼らの作成する計画書、実績報告書を毎週迅速にチェックし（2週間に1回の課題を与えるとすれば計画書、実績報告書に分けると毎週1回となる）修正してやることは、教員1人でこれに対応する場合には負荷が過大になることは明らかである。

考えられる対策の一つは学生同士の相互チェックを行わせ、教員はこの結果を最終的に確認するという方法である。授業時間中に正解例を解説し、これに基づいて個々の学生が他の学生の計画書、報告書をチェックするという方法を採用することは可能であり、チェックする学生の理解を深めるという効果も期待できる。

もう一つの対策は補助指導員の活用である。札幌大学では1991年度から、学生たちの夜間の自主的学習を支援するための補助指導員制度を導入しているが、4年次在籍の最優秀学生を補助指導員に採用する制度には多面的な効果があることが確認されている。この制度をさらに拡大し、補助指導員に計画書、実績報告書のチェックを行わせることは、有力な対策となり得る。

負荷増大への対策がなされれば、残る問題は Know-How の蓄積である。

周到緻密な計画に基づく業務推進の方法が極めて重要であり、プログラミング教育を通してこれを教育し得ること、またこれによりプログラミング教育の学習効率自体が画期的に改善されるであろうことは明らかである。これを実績で示すことが文科系大学における今後のプログラミング教育に寄与することを

信じ、努力を重ねる所存である。

謝 辞

1992年12月に札幌学院大学で開催された情報処理教育研究会は内容的にも充実したものであり、また私自身にとっても日頃の情報処理教育を振り返る貴重な機会になった。このような機会をお作りくださった札幌学院大学社会情報学部の皆様方の熱意と努力に心からお礼申し上げるとともに、敬意を表するものである。

文 献

- (1) J. ハーリー：減びゆく思考力，大修館書店（1992）。
- (2) 松本 吉弘：ソフトウェア工学演習，P. 50，朝倉書店（1984）。
- (3) 安西祐一郎：問題解決の心理学，中公新書（1985）。