

# 文科系学科におけるコンピュータサイエンス教授法 —データベース教育を事例にして—

河 村 一 樹†

コンピュータサイエンスは、理論・抽象化・設計の3つの領域を、系統的に網羅した学問体系である。文科系学科におけるコンピュータサイエンス教育は、なかなか浸透していないのが現状である。この原因には、理数科嫌いの学生があげられる。数理科学をベースにした基礎理論や抽象的な概念について、理解できずに拒否反応を起こすためである。著者の所属する文科系学科におけるデータベース教育（前期に「データベース概論」を、後期に「データベース技法」を実施）においても、数年前からこのような問題が顕著になった。そこで、新たな改善策として、前期に「データベース技法」を、後期に「データベース概論」を実施するように変更した。演習を通して具体的な技術を経験したうえで、講義でそれらの技術の基盤になっている概念や理論を展開することにした。その結果、学生の理解力が向上し、途中での履修放棄がなくなった。以上より、文科系学科におけるコンピュータサイエンスの教授法としては、講義だけでなく演習も併設すること、演習と講義の関連を明らかにすること、演習から講義へ「技術（具体化）から理論（抽象化）へ」という展開が適していることが分かった。

## A Method of Teaching Computer Science in a Department of Liberal Arts — As an Instance of Database Education —

KAZUKI KAWAMURA†

Computer science as an academic discipline systematically includes three domains: theory, abstraction, and design. Computer science education has not yet permeated departments of liberal arts. One of the reasons for this is that students are prejudiced against science subjects, tending to avoid learning about basic theories and abstract concepts based on mathematical science. Some problems of this type concerning database education arose in our department. We therefore revised the course of lectures by adopting a new approach. In concrete terms, "Database Techniques" was assigned as a subject for the first half-year, and "Introduction to Databases" for the second half-year. This approach emphasized the evolution from concretion to abstraction, thus greatly improving the students' understanding of computer science. In summary, we have established a new approach - from technique (concretion) to theory (abstraction), or from practice to lecture - as a method for teaching computer science in departments of liberal arts.

### 1. はじめに

大学などの情報専門系学科におけるコンピュータサイエンス教育は、米国のカリキュラム<sup>1)</sup>をはじめ、我が国のカリキュラムも含めて<sup>2)</sup>、普及し定着した観がある。

一方、情報専門系でない文科系学科におけるコンピュータサイエンス教育はなかなか浸透していない<sup>3)</sup>のが現状である。この原因には、学生の理数科嫌いがあげられる。理数系が弱いために、基礎的な理論や抽

象的な概念の理解に拒否反応を起こすことが多い。この結果、操作技能の習得を目的にしたコンピュータリテラリー教育ばかりが行われている。

しかし、文科系学科も含めた高等教育機関が担う情報処理教育は、単なる操作教育ではなく、系統的な学問としてのコンピュータサイエンス教育であるべきである。そのためには、各専門分野ごとに基礎理論から応用技術に至るまでの一貫した学問体系（理論・抽象化・設計領域）を教授する必要がある。

データベースに関する分野は、学問としての歴史も比較的長く、コンピュータサイエンスの基礎的な領域である理論面からモデル化そして実際の応用技術面に至るまでの学問体系が、系統的に確立されていると

† 尚美学園短期大学情報コミュニケーション学科

A Course of Information Communication, Shobi Junior College

いってよい。このため、著者の所属する文科学科においても、それらの成果を取り入れた教育を、平成3年度から3年間実施してきた。具体的には、基礎理論から応用技術へという流れに対応する形で、講義から演習へという順序で講座を実施してきた。前期が講義を中心にした「データベース概論」で、後期が演習を中心にした「データベース演習」である。

しかし、いくつかの問題が生じてきた。最も顕著な問題は、講義に対する理解力の低下とそれにとまなう履修放棄の増大である。文科系の学生が中心になるため、数学をベースにしたデータベースの基礎理論（集合論、関係代数論、正規化理論など）の説明に拒否反応を起こし、途中であきらめてしまう姿勢が見られた。このため、講義と演習が関連づけられず、後期の演習科目を履修しない学生が増えてしまい、データベース利用技能の習得までに至らない学生が多数生じてきた。

そこで、新たな改善策として、いままでとはまったく逆の展開に基づくカリキュラムを考え、平成6年度から実施を始めた。それは、科目の開講順序を、講義から演習ではなく、演習から講義に全面的に変更したことである。具体的な技術から抽象的な理論へといった展開をはかることによって、分かりやすさを重要視するという方策を打ち出している。それとともに、演習で取り上げた具体的な事例を用いながら基礎理論を説明することによって、その理解力が向上することが分かった。

本稿では、まず他の既存のカリキュラム例を参照する。そのうえで、本学科独自のデータベース教育における到達目標を明らかにする。次に、本学科でのそれまでのカリキュラム内容とその問題点、そして、具体的な改善策とその評価について述べる。

以上のような5年間に及ぶデータベース教育を通して、文科系の学生に対して、基礎理論から応用技術まで含めたコンピュータサイエンス教育を実施するためには、技術（具体化）から理論（抽象化）へという流れのもとに実施した方が教育効果があがることを提案する。

## 2. 他のデータベース教育のカリキュラム例

コンピュータサイエンスをベースにしたデータベース教育に関するカリキュラムには、ACM などによるカリキュラム<sup>4),5)</sup>がある。その中のプロトタイプカリキュラム<sup>4)</sup>の中に、「データベースと情報検索」に関する項目が提示されている。それによると、次のような3つのパラダイムに対応した事項が列挙（ただし、抜粋）されている。

### ① 理論

- 関係代数および関係算術
- 依存性理論
- 並列性理論
- 統計的理論

### ② 抽象化

- データモデル（実体関係モデル、リレーショナルモデル）
- ファイル表現（インデックス、木、逆変換）
- 保水性保証
- 照会言語

### ③ 設計

- データベース（システム）設計技術
- 大規模データベースのマッピング技術

以上のカリキュラムで特徴的なことは、データベースに関する基礎理論を前提に、モデリングそして実装技術に至るまで一貫して体系づけている点（理論・抽象化・設計）にある。コンピュータサイエンスをベースにするということは、このような学問領域から応用領域まで含めた幅広い系統的な視点のもとに、教育を展開することでもある。

## 3. データベース教育の到達目標

情報処理教育におけるデータベース教育の目標については、いくつかの文献で指摘されている。たとえば、大学の情報学科におけるカリキュラム「情報管理・データベース論」<sup>6)</sup>では、手順向き情報処理より対象向き情報処理に重点を置いた教育としてデータベース教育を位置づけている。それとともに、対象向きデータ記述機構の習得を目標に捉えている。あるいは、通産省の高度情報処理技術者の中のテクニカルスペシャリスト（データベース）育成カリキュラム<sup>7)</sup>では、データモデル化技術とデータベース技術の習得を、その教育目標にしている。

これらを参考に、本学科のデータベース教育では、次のような2つの目標をあげている。

**【第1目標】** 外界の事象をモデル化して、データベースに実装できること

データベース化の対象となる情報を、何らかのモデルによって抽象化する必要がある。そのために、さまざまな概念モデルや論理モデルが提案されている。

概念モデルには、E-Rモデルや意味モデルやオブジェクト指向モデルなどが含まれる。論理モデルには、階層モデルや網モデルや関係モデルやハイパーモデルなどが含まれる。これらの各データモデルを理解したうえで、外界の事象を構成する情報群を原始データに

変換する。そして、それらの原始データを、コンピュータの記憶媒体上（磁気ディスクか光ディスク）にデータベースとして格納する。

以上の一連の手順を踏むためには、2章における「理論」と「抽象化」の部分を理解することが重要な鍵になる。

たとえば、関係データベースの場合は、スキーマモデルそのものが厳密な関係代数から演繹されているという経緯がある。したがって、特に関係データベースに関しては、基礎理論からデータベースモデルを捉えると系統的な理解が可能になる。

**【第2目標】** 実装したデータベースから、自由自在に情報を取り出し加工できること

データベースに格納された情報を効率よく取り出すとともに、さまざまな形にアレンジして付加価値をつける必要がある。そのために、さまざまなデータベース言語が提案されている。

自立言語方式として独自の構文体系を持つデータベース言語や、親言語方式としてホスト言語とのインタフェースを持つデータベース言語などがある。ただし、後者に関してはプログラム言語とのインピーダンスミスマッチが生じる場合がある。また、最近では、ハイパーテキストに準拠したスクリプト言語や、ノンプログラミングが可能な第4世代言語（データベースアクセス機能を包括）もある。

これらのデータベース言語を用いて、データベースのアクセスを行うためには、2章における「設計」の部分を理解することが重要な鍵になる。

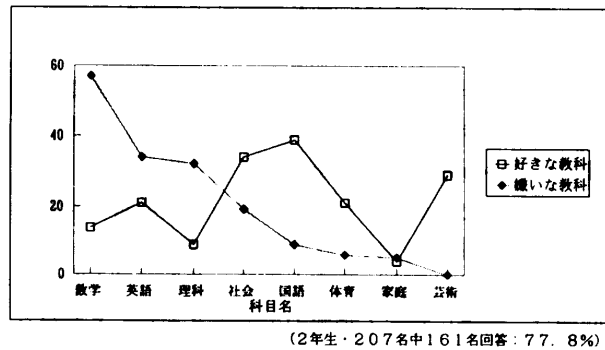
これについても、関係データベースが提供するデータベース言語（SQL）は、関係代数によってモデル化された演算機構に基づいたデータベース操作が可能である。したがって、単にデータベース製品が提供している言語仕様を理解するのではなく、その基盤にある「設計」をベースにして把握することによって、目標を達成することができる。

#### 4. データベース教育の実際

ここでは、本学科におけるデータベース教育の変遷について、いままでのカリキュラムとその問題点、および、改善策について明らかにする。

##### 4.1 受講対象になる学生

筆者の所属する情報コミュニケーション学科は、平成2年に設置された社会科学系の学科である。このため、どちらかという文科系に近い位置づけになることから、学生の多くは数学に対してアレルギーを持っている。これについては、図1から見出すことがで



(2年生・207名中161名回答: 77. 8%)

図1 高校での得意/苦手科目

Fig. 1 Skillful/weak subjects at a senior high school.

きる。

##### 4.2 いままでのカリキュラム

コンピュータリテラシーに関しては、1年次に「コンピュータ基礎演習」(必修, 1単位)と「コンピュータ演習」(必修, 1単位)などの演習科目を開講している。このため、ワープロや表計算や作画やパソコン通信といった基本的操作は、それぞれできるようになっている。なお、データベースに関しては、表計算ソフト (LOTUS1-2-3) の中で簡単な演習を行っている。

コンピュータの基礎的な理論や専門知識に関しては、1年次に「コンピュータシステム論」(必修, 2単位)や「コンピュータ科学論I」(必修, 2単位)などの講義科目を開講している。

以上によって、1年次までに、コンピュータ全般についての基礎的な知識と技術を習得できるようなカリキュラム体系になっている。それらを基盤にしたうえで2年次に、より専門分野に細分化された領域 (データベース, ネットワーク, 応用ソフトウェア) を習得できるようなカリキュラム体系になっている。

なお、本学科は社会科学系であり、コンピュータ以外の科目が中心になっている。このため、コンピュータ関連科目全体としては、3コマ程度/週しか、時間をとることができないという制約が生じている。

以上より、本格的なデータベース教育は、2年次になってから実施することになる。2年次前期には「データベース概論」(選択, 2単位)を、後期には「データベース演習」(選択, 1単位)を、それぞれ開講した。科目ごとの概要については、それぞれのシラバスを表1と表2にまとめる。

##### 4.2.1 (旧) データベース概論

「データベース概論」については、講義が中心になる科目である。

前半の基礎理論 (第2回~第4回) については、次の事項について、数式を用いた説明が中心になる。

表1 (旧)「データベース概論」のシラバス

Table 1 A syllabus about "(old) Introduction to Databases".

回数	取り上げるテーマ
1	データベースの歴史の変遷と今日のデータベース利用
2	データベースの基礎理論 (集合論)
3	データベースの基礎理論 (関係代数)
4	データベースの基礎理論 (正規化理論)
5	データモデル (概念モデル, 論理モデル)
6	データベース実装技術 (物理モデル, 編成法)
7	データベース管理システム (機密保護, 排他・障害制御)
8	スキーマモデル (データ独立性, 3層スキーマ)
9	関係データモデル (関係スキーマ, 候補キー, 一貫性制約)
10	関係データベース言語 SQL (発展経緯, 標準化動向)
11	スキーマ定義言語 (SQL-DDL)
12	データ操作言語とプログラムインタフェース
13	新データベース (オブジェクト指向, マルチメディア)

表2 (旧)「データベース演習」のシラバス

Table 2 A syllabus about "(old) Database Techniques".

回数	取り上げるテーマ
1	E-R モデルの設計演習
2	正規化によるスキーマモデルの設計演習 (1)
3	正規化によるスキーマモデルの設計演習 (2)
4	関係データベース言語 SQL のコマンド体系
5	SQL-DDL (CREATE SCHEMA) による実装演習
6	SQL-DDL (CREATE TABLE) による実装演習
7	SQL-DDL (CREATE VIEW, GRANT) による実装演習
8	SQL-DML (問合せ) 操作演習
9	SQL-DML (非カーソル問合せ) 操作演習
10	SQL-DML (非カーソル更新) 操作演習
11	SQL-DML (カーソル) 操作演習
12	SQL プログラミング演習
13	総合演習課題

## ① 集合論

- 集合の定義
- 部分集合
- 集合則
- 集合演算 (和/差/共通/直積)

## ② 関係代数

- 関係の定義
- 関係演算 (射影/選択/結合/商演算)
- 演算の等価性

## ③ 正規化理論

- (完全・推移的) 関数従属性と第1/第2/第3正規形 (1/2/3NF)
- ボイスコッド正規形 (BCNF)
- 多値従属性と第4正規形 (4NF)
- 結合従属性と第5正規形 (5NF)

文科系の学生であるため, 最初は高校時代の数学の復習を兼ねた形で, 講義を進めなければならない。それでも, 数学に対する拒否反応が強いようで, データ

ベース以前の対策が必要になる。このため, 別途演習問題を配布し, 授業時間外に個々人で演習させる必要がある。

第5回から第9回までは, データベースのモデル化の世界について説明することになる。このため, 話しの内容が非常に抽象的になりやすいので, 具体的事例をいくつか提示したり, 図式を多用した説明を行わなければならない。

第10回から第12回までは, SQLの構文を中心に取り上げる。ただし, あくまで「データベース演習」で演習ができるように, 具体的な例を用いながら構文規約を主に説明する。

第13回では, データベースの展望として, 次世代データベースの話題について説明する。

## 4.2.2 (旧) データベース演習

「データベース演習」については, 演習が中心になる科目である。

講義で習得した内容をもとに, コンピュータ (パソコン) にデータベースを実装したうえで演習を行うことになる。これによって, 理論から技術への系統的なデータベースの学問体系を展開することができる。また, 3章で取り上げた到達目標を実現するうえで, 最も適した科目に相当するといつてよい。

その演習<sup>8)</sup>には, パソコンで稼動するデータベース管理システム製品である Informix-SQL<sup>9)</sup>および Informix-ESQL/C<sup>10)</sup>を用いる。前者は, 関係データベースを実装するための製品であり, 標準SQL<sup>11)</sup>を網羅している。これによって, 会話形式で関係データベースの定義や操作の演習を実施することができる。後者は, SQLのプログラムインタフェースを支援するための製品である。これによって, C言語のプログラムにSQLを埋め込むことができ, SQLプログラミング<sup>12)</sup>の演習を実施することができる。

第1回では, E-Rモデルを用いて, 対象の問題領域をモデル化する演習を行う。講義では, E-Rモデルの概要を取り上げているので, 実際の例を見せながら演習課題を与える。

第2回から第3回までは, 正規化理論に基づく実際の演習を行う。ここで, 数式で定義した正規形を実際の事例とともに解説する。そのうえで, 非正規形の関係表がある演習課題を与え, 机上で正規化を演習させる。

第5回以降は, SQL-DDLおよびSQL-DMLにおけるコマンドの構文規約とその意味を習得するための演習を行う。

SQL-DDLによる実装演習 (第5回~第7回) で

は、正規化した関係表をもとに、次のような演習課題を行う。

- スキーマ認可識別子の一意性に関する課題
- 属性定義におけるデータ型宣言に関する演習
- 一貫性制約（キー制約，参照制約，検査制約，非空白値制約）に関する課題
- ビュー表における参照制約/更新制約に関する課題
- 権限付与に関する課題

SQL-DMLによる操作演習（第8回～第10回）では、実装した関係表をもとに、次のような演習課題を行う。

- 単純問合せ（関係演算，集計関数，グループ演算）に関する課題
- 結合問合せに関する課題
- 副問合せに関する課題
- 更新（挿入，変更，削除）に関する課題

このうち、問合せの課題の中で、SQLのSELECT文による関係演算（射影，選択，結合）の構文と、講義で取り上げた基礎理論の展開との関連について明らかにする。これによって、基礎理論と応用技術とのつながりが明確になる。

カーソルを含めたSQLプログラミングによる演習（第11回～第12回）では、会話形式のコマンドによる演習だけでなく、C言語を用いたSQLプログラミングについても演習を行う。ただし、プログラムの枠組みを与え、そこにSQLを埋め込むようなプログラム演習を実施する。

- 非カーソルによる埋込みSQLプログラミング
- カーソルによる埋込みSQLプログラミング

第13回では、最終的な総合演習課題を個別に与える。自分独自の関係データベースの設計および実装を行わせるとともに、その利用方法についてレポートさせるという課題である。その結果として、最も多いのが名簿データベースであった。それ以外では、料理データベースや飲食店データベースや音楽データベースなどがあげられる。

#### 4.3 データベース教育の問題点

4.2節のようなデータベース教育を3年間にわたって実施してきたが、いくつかの問題点が顕著になってきた。これは、履修者数の推移や授業に対する学生のアンケート調査結果などから明らかになったわけである。

以下に、その問題点を具体的に列挙する。

##### 4.3.1 「データベース概論」の履修放棄

表1で示したように、講義では基礎的な理論や抽象的な概念についての説明も多くなる。このため、数式や図式を用いた説明が続くことになる。1年次に数学

を学ぶ機会は設定されているが、社会科学系の学科であるため、数学に対するアレルギーが強い学生も多い。また、データベースのモデルに関する部分に関しても、ある程度の抽象化能力がなければ理解しにくい。

このような学生に対して講義を続けていくと、必然的に学生のやる気が減退してしまうといった状況が発生してくる。いくら懇切丁寧に講義を行っていても、学生の理解力が低下するという傾向が顕著になってくる。この結果、授業途中で履修放棄が出てきてしまう。

##### 4.3.2 講義と演習の遊離

前期でデータベースに関する理論や概念を知識としてきちんと理解させてから、後期の演習においてそれらを技術として習得させるという教育過程を想定していた。ここでは、講義と演習が有機的に関連づけられながら、知識が技術に変換されるという習得プロセスを前提にしていたわけである。

しかし、学生から見ると、「データベース概論」と「データベース演習」はまったく異なった科目であり、単位取得のためだけになっていた。このため、前期の履修だけで、後期の履修を行わない学生が出てきた。また、「データベース概論」はわけの分からない難解な講義であり、「データベース演習」は単にパソコンを操作するための演習であるとしか見ない学生もいた。この結果、演習の方では、操作の習得だけが主な目的になってしまい、普遍的な技術にまで一般化することができなくなる。したがって、別のデータベースソフトは使えないと主張する学生も出てきた。

##### 4.4 その改善策

以上の問題は、文科系の学生に対して、いかに分かりやすくコンピュータサイエンスという学問を教授すべきかということに結び付くといってもよい。そのときに、数式や図式をベースにした難解で抽象的な講義は、残念ながらこのような学生には適していないことになる。

そこで、4.3.2項で取り上げた習得プロセスを逆転したパターンを思いついた。つまり、実際の技術から基礎理論を導き出すというアプローチである。コンピュータを実際に操作し、具体的な事例に慣れさせたうえで、それらを生み出した根源である抽象的な理論や概念的な知識を紹介するという手順である。文科系の学生に対しては、まずデータベースの諸機能を実体験させ、具体的なイメージを把握させる。そのうえで、演習課題と関連づけながら概念や理論について説明すると、教育効果が高くなるといえる。

抽象化とは、本来その対象になる事象から代表的な

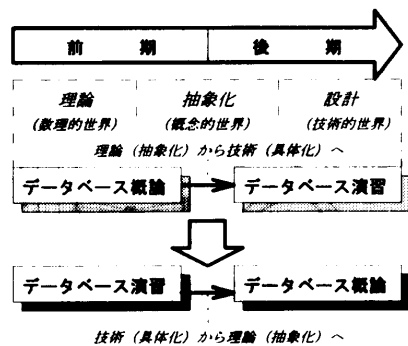


図2 改善策の構成

Fig. 2 A construction of improvement approach.

部分だけを抽出し、それ以外の事象を捨象することによって実現できる。そのためには、事象を構成する多くの具体的な要素を洗い出す作業が必要になる。そのうえで、特徴的な性質の要素だけをまとめてモデル化することになる。これは、「具体化から抽象化へ」の展開にも相当する。

以上の考え方に基づいて、いままでの習得プロセスを逆転し、前期に「データベース演習」を、後期に「データベース概論」を開講することにした。図2に、この関係を示す。

なお、前期(1コマ分)と後期(1コマ分)の科目を一本化して一つの科目(2コマ分)として開講することも考えられたが、コンピュータ関連科目に配当される時間数が少ないことから、1つの科目だけを2コマ/週という形で設置することができなかった。また、データベースという科目の特性から、講義(理論や知識)と演習(技術や実践技能)を分けやすいことがあった。これより、前期と後期の並びで対応することにした。

それとともに、2年次での各科目は、より細分化された専門領域に位置づけていることから、該当科目だけで独立してカリキュラムの再編を試みた。したがって、カリキュラム全体からの見直しまでは考慮せずに進めることを、本学科において了承したという経緯がある。

以上の形での開講にともない、科目の内容を一部変更することにした。科目ごとの概要については、それぞれの改訂後のシラバスを表3と表4にまとめる。

#### 4.4.1 「データベース演習」に関する改訂

演習科目から入るので、学生はデータベースに関する知識や専門用語などについてはほとんど知らない。そこで、第1回にデータベース全般の概説(利用目的や技術背景)、および、関係データベースのスキーマ構造についてだけ講義する必要がある。そのうえで、演習に入る。

表3 (新)「データベース演習」のシラバス

Table 3 A syllabus about "(new) Database Techniques".

回数	取り上げるテーマ
1	データベース全般および関係データベーススキーマの概説
2	正規化によるスキーマモデルの設計演習(1)
3	正規化によるスキーマモデルの設計演習(2)
4	関係データベース言語SQLのコマンド体系
5	SQL-DDL(CREATE SCHEMA)による実装演習
6	SQL-DDL(CREATE TABLE)による実装演習
7	SQL-DDL(CREATE VIEW, GRANT)による実装演習
8	SQL-DML(問合せ)操作演習
9	SQL-DML(非カーソル問合せ)操作演習
10	SQL-DML(非カーソル更新)操作演習
11	SQL-DML(カーソル)操作演習
12	SQLプログラミング演習
13	総合演習課題

表4 (新)「データベース概論」のシラバス

Table 4 A syllabus about "(new) Introduction to Databases".

回数	取り上げるテーマ
1	関係データモデル(関係スキーマ、候補キー、一貫性制約)
2	関係データベース言語SQL(発展経緯、標準化動向)
3	スキーマ定義言語(SQL-DDL)
4	データ操作言語(SQL-DML)
5	プログラムインタフェース
6	スキーマモデル(データ独立性、3層スキーマ)
7	データベース実装技術(物理モデル、編成法)
8	データモデル(概念モデル、論理モデル)
9	データベース管理システム(機密保護、排他・障害制御)
10	データベースの基礎理論(集合論)
11	データベースの基礎理論(関係代数)
12	データベースの基礎理論(正規化理論)
13	新データベース(オブジェクト指向、マルチメディア)

第2回～第3回の正規化に関する演習については、正規化という用語を用いず、いくつかの問題について考えさせればよい<sup>3)</sup>。たとえば、次のような第3正規形ではない関係表を与える。

生産表(生産者コード、生産品コード、生産高、所属組合、生産地区)

※ 候補キー：生産者コードと生産品コード

この関係表をもとに、更新時異状および情報無損失分解<sup>13)</sup>について説明し、どのようにすれば更新時異状が起きなくなるのかについて、机上で演習させる。以上のことが、正規化理論に結び付くわけだが、ここでは例題をもとに専門用語を用いず演習を進める。

第4回以降は、(旧)「データベース演習」のシラバスと同様になる。

#### 4.4.2 「データベース概論」に関する改訂

「データベース演習」では、Informix-SQLを用いたスキーマ定義とデータ操作およびプログラミングの演習が中心であった。

「データベース概論」では、演習で習得した実践的な技術を、その基盤となる系統的な知識体系や基礎理論に抽象化することを目的とした講義が主になる。すでに、演習課題を通して具体的な意味については学生が理解しているので、教授する側にとっても講義しやすい。演習で与えた課題を講義の際に利用することもできる。一方、学生にとっても、与えられた課題をもとに講義が進むので理解しやすいことになる。

改訂にあたっては、(旧)シラバスに対してその順番をほぼ逆転したものになった。つまり、技術からモデリングそして理論へとさかのぼるアプローチをとる。以下には、変更点だけを列挙する。

第1回では、関係表を単なる二次元の表としてでなく、データモデルという視点から厳密に定義する。これによって、関係スキーマの不変性と、タプルの集合体としての関係インスタンスの変化について明らかにする。

第2回から第5回までは、SQLの構文と言語体系について、系統的な視点から説明する。シラバスのテーマは、旧のものと同様であるが、ここではBNF記述による構文記述<sup>11)</sup>を用いる。これによって、BNFによる仕様記述の方法についても理解させることができる。

第10回から第12回までは、基礎理論であり、旧カリキュラムでは、最も問題のあった部分である。しかし、「データベース演習」を履修したことから、演習で行った課題をもとに説明ができる。具体的には、次のようになる。

集合論については、(新)「データベース演習」の第5回および第6回の演習課題を用いることができる。ここで、集合が関係の定義に用いられることを示す。つまり、直積を用いて関係を厳密に導くことができるわけである。そこで定義された関係が、二次元の表として表現できることを明らかにする。

関係代数については、(新)「データベース演習」の第8回の演習課題を用いることができる。ここで、SQLのSELECT文による関係演算(射影、選択、結合)と集合演算(和、共通、差)が、それぞれ代数式として定義できることを示す。

正規化理論に関しては、(新)「データベース演習」の第2回および第3回の演習課題を用いることができる。ここで、初めて正規化という用語を用いて、正規化のための基礎理論を代数的に説明する。

以上によって、関係データベースが厳密な数理理論から生み出されていることを明らかにすることができる。

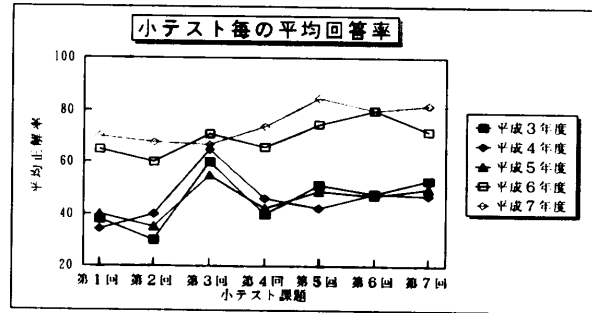


図3 科目の評価

【課題】第1回：集合論に関する計算問題、第2回：非正規形データを3NFに正規化する問題、第3回：RDBのデータ独立性の意義を述べる問題、第4回：関係代数とSQLの関係を定義する問題、第5回：関係表の定義例をSQL-DDLに変換する問題、第6回：関係表の検索例をSQL-DMLに変換する問題、第7回：SQLプログラミングの穴埋め問題

Fig. 3 An example estimation of subject.

#### 4.5 改善した科目に対する評価

以上の改善に基づくデータベース教育を、平成6年度から実施した。その結果、改善前後では、以下のような点に効果が見い出された。

##### (1) 履修放棄の減少

改善前では、前期の「データベース概論」で約20%の履修途中での放棄が見られた。また、後期の「データベース演習」を続けて履修する者は、約70%しかいなかった。

改善後では、両科目とも履修途中での放棄がまったく見られなくなっている。また、前期「データベース演習」から後期「データベース概論」への履修継続は、全員になっている。

このことから、まずはコンピュータを用いてデータベースを操作することのおもしろさを感じ取っているといえる。それとともに、講義を通してデータベースに関する体系的な知識を習得する意欲が見られる。

##### (2) 基礎理論の理解度の向上

改善前と改善後の教育効果の比較を行うため、それぞれの科目の中で、同一の課題による小テストを実施してきた。その結果を、図3に示す。

これによると、改善を実施した平成6年度から各課題の平均正解率が、著しく向上していることが分かる。特に、数学に関連した課題(第1回、第2回、第4回)に対する正解率がいずれもあがっている。

これより、4.1節で述べたような文科系の学生に対して、数式だけを用いて基礎理論を講義することは適していないことが分かる。とにかく具体的な事例(自分で実習した演習課題など)を用いることによって、基礎理論を結び付けていく必要がある。具体化から抽

象化のアプローチが有効であることが分かる。

### (3) 到達目標の確実性

改善前では、いずれも単位取得のための科目とみられがちであった。つまり、科目を履修しても受動的に学習しているだけで、自分でデータベースを利用してみるという姿勢はほとんどの学生が持っていなかった。

改善後では、学生が自分の卒論執筆のためのデータ収集と解析に Informix-SQL を用いたり、情報コースの学生の中には Informix-ESQL/C を用いて関係データベースのフロントエンドプロセッサを開発するテーマを卒業制作に選んだりするようになってきている。これらは、いままでにはなかった現象である。

### (4) コンピュータサイエンスへの関心の高揚

本稿で取り上げてきたデータベース教育は、コンピュータサイエンスをベースにしている。したがって、単なる専門知識の羅列や技能習得に陥ることなく、系統的な学問領域からの理解に重点を置いている。そのために、データベースに関する基礎理論についても取り上げているわけである。

以上のような姿勢が学生にも波及効果を与えているようで、学問としてのコンピュータサイエンス領域に関心を持ち始めている。このため、コンピュータサイエンスの基礎を取り上げる「コンピュータ科学論 II」の履修者数も増加している。

## 5. おわりに

本稿では、文科系学科におけるコンピュータサイエンスをベースにしたデータベース教育について述べてきた。その中で、抽象的な理論と具体的な技術との関連性を学生に明確に把握させるためには、理論から技術よりも、技術から理論の方が有効であることが実証された。つまり、具体化から抽象化へ展開するカリキュラムの確立である。特に、数学に対するアレルギーの強い文科系の学生にとっては、このような教授法が非常に効果的であることが明らかになった。

また、このような形で、コンピュータサイエンス教育を行うことの重要性を確認することができた。高等教育機関であるからには、学問的な立場をもとに教育を行う必要がある。そのためには、いかにうまくコンピュータサイエンス教育を実施するかが課題になる。ここでは、情報専門系でない学科において、コンピュータサイエンス教育を効果的に進める教授法の一例として提案した。

以上のことは、他の教育機関にも適用できると思われる。たとえば、今後より普及が進んでいくと思われる初等・中等教育機関において、コンピュータサイエ

ンスの入門的なカリキュラムを確立する際にも応用できる。特に、コンピュータの初心者にとって、拒否反応を与えないように教育を進めていくため、具体的な利用から入ることは効果的である。それだけでなく、そこで習得した技術を基礎的な知識や理論にレベルを合わせながら教授する学習指導要領を作成することもできる可能性がある。

また、(文科系が多いような) 企業の新入社員教育などでも、最初から講義一辺倒ではなく、演習を前倒しにしたカリキュラム編成が考えられる。受講生の理解度を高める手段として、活用することができる。

今後の課題は、データベース教育だけでなく、他の科目においてもコンピュータサイエンス教育を実施するための具体的な方策について検討を進めたい。

## 参考文献

- 1) 国友利泰 (編)：コンピュータサイエンスのカリキュラム，共立出版 (1993)。
- 2) 大学などにおける情報処理教育のための調査研究委員会 (編)：大学などにおける情報処理教育のための調査研究報告書，情報処理学会 (1991)。
- 3) 短期高等教育機関における情報処理教育の実態に関する調査研究委員会：短期高等教育機関における情報処理教育の実態に関する調査研究報告書，情報処理学会 (1995)。
- 4) Denning, P.J., Comer, D.E., Gries, D., Mulder, M.C., Tucker, A., Turner, A.J. and Young, P.R.: Report of the ACM Task Force on the Core of Computer Science, ACM Press (1988)。
- 5) Tucker, A.B., Barnes, B.H., Aiken, R.M., Barker, K., Bruce, K.B., Cain, J.T., Conry, S.E., Engel, G.L., Eptein, R.G., Lidtke, D.K., Mulder, M.C., Rogers, J.B., Spafford, E.H. and Tuener, A.J.: Computing Curricula 1991 - Report of the ACM/IEEE-CS Joint Curriculum Task Force, ACM Press (1991)。
- 6) 坂井利之ほか (編)：情報工学の教育・研究，共立出版 (1980)。
- 7) 通産省機械情報産業局 (編)：ソフト新時代と人材育成，通産資料調査会 (1993)。
- 8) 河村一樹：パソコンによる関係データベース言語 SQL の演習教育について，第 41 回情報処理学会全国大会論文集 (1990)。
- 9) Informix-SQL Reference Manual I/II, ASCII (1992)。
- 10) Informix-ESQL/C Programmer's Manual, ASCII (1992)。
- 11) JIS データベース言語 SQL (JIS X3005-1990)，日本工業規格協会 (1992)。
- 12) 原 潔：標準 SQL プログラミング，啓学出版



- (1990).
- 13) 増永良文：リレーショナルデータベース入門，サイエンス社 (1991).
- 14) Date, C.J.: *An Introduction to Database Systems*, 3rd edition, Addison-Wesley (1981).
- 15) Celko, J.: *SQL for Smarties: Advanced SQL Programming*, Morgan Kaufmann Publishers (1995).
- 16) Date, C.J.: *A Guide to THE SQL STANDARD*, Addison-Wesley (1987).
- 17) 植村俊亮：データベースシステムの基礎，オーム社 (1979).
- 18) 河村一樹：データベース要論—関係データベースとオブジェクト指向データベース—，ダイゴ (1995).
- 19) 河村一樹：SQL 言語活用入門，日刊工業新聞社 (1990).
- 20) 河村一樹：理論と実際を関連付けたコンピュータサイエンス教授法—データベース教育を事例にして—，情報処理学会研究報告，Vol.95, No.75, pp.17-24 (1995).
- 21) 河村一樹：文科系学科におけるコンピュータサイエンス（基礎）教育のあり方について，教育システム情報学会誌，Vol.13, No.1, pp.44-50 (1996).

(平成7年12月13日受付)

(平成8年9月12日採録)



河村 一樹 (正会員)

1955年生。1978年立教大学理学部化学科卒業，1989年日本大学大学院理工学研究科博士前期課程電子工学専攻修了。現在，尚美学園短期大学情報コミュニケーション学科助教授。情報処理学会情報処理教育カリキュラム委員会委員。情報処理学会コンピュータと教育研究会幹事。CAIT 第一種共通カリキュラム研修講師。主として，情報処理教育の研究に従事。電子情報通信学会，教育システム情報学会，日本教育工学会各会員。