

解説



情報処理専門教育について

情報処理専門教育の一実例

—大学における情報工学教育の理念と実現†—

小谷善行^{††} 高橋延匡^{††}

1. はじめに

東京農工大学旧数理情報工学科では十数年来、独自の情報工学教育を立案し実施してきた。その目的は、真に情報工学（ないしは計算機工学）を教育研究することであり、自立的な専門的研究者・技術者の育成を目指してきた。これを実現するために提起した理念や仕組みによって当学科は特色のあるカラーをもつことになった。その内容や評価を具体的に述べる。

東京農工大学工学部数理情報工学科は昭和 51 年に 4 講座（職員 17 人）、学生定員 40 人の規模で発足した。その後、平成元年には、博士課程設立にともない、工学部全体の改組を行い、当数理情報工学科に加え、電気・電子・応用物理系の 3 学科とともに合体して電子情報工学科を設立した。この時点で数理情報工学科という名前は消えたが、その体制は同学科コンピュータ・サイエンスコースに引き継がれている（以下コンピュータ・サイエンスを CS と略記する）。現在このコースの学生は 1 学年あたり 50~60 名である。その教育内容の骨格や理念は、ほとんど旧数理情報工学科の時期にできあがった。ここでは、この旧数理情報工学科および、電子情報工学科 CS コースの専門教育を論じる。以下で両者を区別せず「当学科」で表す。

2. 専門教育の理念

ここでまず情報工学（ここでは CS と区別しない）という言葉の位置付けておこう。情報工学とは、計算機にかかわるすべての工学・科学であ

る。計算機自身のハードウェア・ソフトウェア（以下ハード、ソフトと略記）など、本情報処理学会のカバーする分野と考えてよい。日本語では、情報工学（科学）や計算機工学（科学）などいくつかの言葉がある。ここではこの学問分野を CS と呼ぶことにする。つけ加えれば、たとえば電子工学や応用数学は、CS とは近いが異なる学問である。当学科が設立された時期は、まだ CS が日本では一つの分野として認知されていなかった。そのなかで、「ほんとうに計算機や計算機の科学をやる CS の学科を作る」という理念で当学科を設立し、育成してきた。

一方その実現手段となる考えは多様であり、詳細に文書化してあるものは少なくまた教官個人ごとの違いもあり、的確に述べるのは困難である。ここでは次の重要なものと認識している項目、「synthesis の重視」、「not teaching but educating」、「FORTRAN の運転手は作らない」、「CS 分野のバランス」、「ブラックボックスを廃す」、「手作り」、「実物教育」、「パワーエデュケーション」にまとめる。

2.1 Synthesis の重視

CS という学問はその周囲の学問と違う特徴もっている。その一つが synthesis の色彩が強いことである。対象を分析してその構造を明らかにするのが analysis（分析）であるのに対して、無の状態から要素を組み合わせて全体的構造を作り上げるのが synthesis（総合）である。たとえば物理学は対象を解明するのが目的であるとすれば前者を指向した学問である。CS はつねに系を設計することが背景にあるので後者が中心である。もちろん効率分析など analysis 的側面もある。synthesis 重視の考えはおもに実験や演習のなかで現われる。2 年生の実験では工学一般の基礎にかかわるので analysis の面が強いが、その後の実験や

† An Example of Technical Education on Information Processing—Philosophy and Implementation of Computer Science Education in a University—by Yoshiyuki KOTANI and Nobumasa TAKAHASHI (Tokyo University of Agriculture and Technology, Faculty of Technology, Department of Computer Science).
†† 東京農工大学工学部電子情報工学科コンピュータ・サイエンスコース

演習ではもの設計や製作の面が強くなる。

一般に、大学教育は analysis に片寄りがちであるが、それは CS の学問の性質上望ましくなく、社会的ニーズにも一致していない。当学科の実験ではプログラムや回路などの「ものを作る」のが中心である。卒業論文や大学院では、ほぼ全員がハードにせよソフトにせよ、あるシステムを設計し製作する。

2.2 Not Teaching But Educating

「教える (teach)」という、教官の知識を生徒に移殖する、という意味が強い。教育の現場では教えることが中心になりがちであるが、それは望ましくないと考える。逆に教えないことによって、能力を育てる。つまり「教」でなく、「育」を目指す。本来「education」には「教える」という意味は含まれていない。

演習では、プログラミング課題が与えられるが、計算機言語はそれ自身を対象として教育しない。実際の問題を解こうとすることから始まる。方法論を教えることはあるが、その先は自分で参考書を読んだり、自分で研究したりすることを求める。また後述のように、英語の輪講で、英語もそれ自身を対象として教育しない。英語という自然言語にせよ、計算機言語という人工言語にせよ、それ自身を対象とする教育は不要と考える。CS という対象を目的とするのが主旨であり、その過程で言語を身につけるのを理想とする。

2.3 FORTRAN の運転手は作らない

「FORTRAN の運転手は作らない」というのは学科の運営当初からのスローガンであった。そして「機械工学科でタクシーやバスの運転手を作るか!」と続く。現在では「C の運転手は作らない」と言い替えたほうがよいだろうが、今もこの考えは変わらない。その意味は、一つには、2.2 と同じく、言語自体の教育は意味がないという点であるが、ほかに少し違うニュアンスもある。つまり、計算機言語の言語教育は「CS の専門教育」としてあまり意味がない。もちろん技術者一般に対して適切な計算機リテラシー教育は必要である。しかし、それを CS の学生に実施することは機械工学科の学生に自動車運転の講習をするようなものである。ひと昔前の CS 教育にあった「とにかくもまず計算機言語教育を」という傾向を、この言葉で皮肉った面もあった。

2.4 CS 分野のバランス

当学科では、計算機のメカニズムの物理的な部分から応用プログラムの利用者インタフェースまでの全体を扱うことを目指した。理由は第一には、CS 全体を教育するべきであると、第二には、研究分野が多様なほうが、相互に新鮮なインパクトを与え合う結果、研究により影響を与えると考えたからである。

計算機の学問では、単純ではあるが、ハードとソフトという区別が第一の区別として重要である。これは単に対象の違いだけではない。ものを設計するにあたっての思考方式が異なる。ソフトとハードの両方について研究教育体制を整えることを重視した。学生には実験のなかで両方が履修されるようにした。

このほかにも、カリキュラムにおいては、計算機アーキテクチャに近い部分からアプリケーションまでを見渡すように考慮した。たとえば、ロボット、知識処理、画像、データ処理、アルゴリズムなど、CS の中の発展性のある重要な分野をカバーするようにした。

2.5 ブラックボックスを廃す

これは「FORTRAN の運転手を作らない」とこと関わる。ある問題に対してプログラミングでできればよい、とはしない。よいプログラミングには、コンパイラの処理の仕組みを理解しているべきである。計算機アーキテクチャを理解しているべきである。計算機アーキテクチャがなぜある形をとるのかを理解するには、素子レベルの時間的・空間的構造を理解する必要がある。

CS の体系はこのように階層的な構造をもつ面がある。CS の教育として、階層の最も下位の素子レベルから、人工知能系などの上位レベルまで、一貫して理解する能力の修得を目指している。このように系の内部にまで立ち入って理解を試みる。また内部に立ち入れなくても、系の振る舞いを測定し、モデルを作るのもこの考えに含まれる。

2.6 手作り・実物教育

教育の過程で、シミュレータや CAI を使って効率化することは否定しないが、実物を用いて教育することは有形無形の意味があると考えている。実物に立脚した知識には強靱さがある。同じ見地から「手作り」を重視している。すでに完成

した機器やソフトを使うのではなく、機器を手作りして教育する。場合によりその機器やソフトを作る過程も教育の一部にできる。

EDSAC（歴史的に有名な初期の計算機）による教育の理念にも関連したものがある。現代の完成した計算機アーキテクチャをそのまま教育に出しても、その本質的な意味を理解しにくい。原始的な計算機により、むしろ、アーキテクチャに関する基本的な尺度を習得し、その後発展した機構（たとえば、スタックや割り込み、アドレス変換）の位置づけが明確にできる。

2.7 パワーエデュケーション

手をかけた密度の高い教育、といった意味での言葉を使っている。たくさんものを教えることではない。教育に手間をかけることに躊躇しないという意味である。CSに教育という要素が多く含まれると考えている。CAI的なものだけではなく、行っている教育全体を研究の対象として捉えている。

3. 実験および演習の実施

この章では具体的な教育の実施方法を述べる。講義の詳細ではなく、われわれの重視する、実験、演習を中心に論じる。当学科では、改組にしたがって、現在ではCSに関する専門教育は3年生から実施する体制になっている。2年生では、電子工学、電気工学、応用物理学、情報工学に共通する、工学の基礎を教える。そのため、旧数理情報工学科時代に比べて、CSの専門教育に関するカリキュラムがしわ寄せを受けている。解決策として、2年生（一部1年生）でCSの専門教育が再び実施できるよう折衝している。

現在これに対処するために、第一に、CSの基礎的教育として重視している部分を、電子情報工学科全体で教育する意義を学科全体に理解してもらい、それを実施するようにした。第二に3年生の段階で可能なかぎりカリキュラムを詰め込んでいる。

そこで次の説明ではこの改組以前のカリキュラムを基にする。課題は、1年生で一般教育、2年生で工学の基礎と専門教育、3年生で専門教育、4年生で卒業論文研究、と進む。学年順に専門教育を解説する。

3.1 1年生

1年生は府中キャンパス（一般教育部と農学部）に通学し、一般教養を中心に学ぶ。専門教育は2年生からで、小金井キャンパス（工学部）で行う。したがって1年生で当学科のカリキュラムは学科創設当初はなかった。しかし、このままでは1年生の帰属意識や士気、目的意識が希薄になると考え、一つの講義を開講することとした。この講義（情報工学概論）ではCSの歴史、現状、今後の発展を紹介するとともに、進学後の勉学に対して展望を与えた。時期は夏休みの初めで集中講義の形でいった。内容は各教官が1コマずつ行い、古い計算機の基板を見せながら歴史を語ったり、「人工知能とは何か」についてチューリングテストを話題にして哲学的議論をしたり、計算機製作過程のビデオを見せたり、計算機に触らせたりした。学生には各講義内容を含むレポートを課した。なお、この講義の直後に、合宿オリエンテーションという教官と学生の親睦の行事を開催した。これには上級生や大学院生の参加もあり、1年生から教官までの全員の一体感が生まれるなど学問以外の点にも意味があった。

この講義は数理情報工学科の講義として約10年続き、その目的を十分満足していたと評価できる。しかし残念ながら、改組以降はこの主旨の講義は不可能になった。電子情報工学科の学生がCSコースに配属されるのは3年からであり、1年生のうち物理や電気や電子や計算機のうちどれを専門とするかは決まっていない。そのため現在の講義（電子情報工学概論）では、CSに関して、4倍の人数に4分の1の内容を教えることとなり、帰属意識や目的意識を与えるという意図は実現できなくなった。

3.2 2年生の演習

基本的に2種類ある。第一には応用数学的なものである。数学の応用というより、工学的見地からの数学である。現象をモデル化するための基本的手段を学ぶ。古典力学、解析力学、電磁気学、複素関数論、ラプラス変換、微分方程式などを含み、演習形式で行う。

第二はCSの基礎である。その一つは、計算機械の講義に関連して、計算機に関する基本的な演習を行う。ここではEDSACのプログラムをみっちり作成する。これで計算機アーキテクチャの

可能性、プログラミングの可能性の根底的な部分を肌身で理解する。もう一つは、スタック、ハッシュ法、ソートなどのソフトの基礎技術・概念の演習を行うものである。これらにはおのおのプログラムの課題が与えられる。前述のように計算機言語自体の教育などは原則としてしない。プログラム課題は問題を解く訓練であり、計算機言語は手段である。このほかにタッチタイプの訓練がある。これは今後の研究生活の効率のためだけでなく、ヒューマンインタフェースに目を開かせるためでもある。

3.3 2年生の実験

これは基本的にはハード系の実験である。多くが工学一般の基礎と共通するものである。しかし、そのなかで独自の工夫がある。第一には、レポートの重視である。第二にはCSの基礎的概念のテーマである。レポートの重視とは、論文や報告書などの科学技術文書としてレポートを考へることである。当学科の作成している実験マニュアルの目的にはこの点を含んでおり、「報告書は答案ではない」、「自分の文章で書く」などの諸注意事項が示されている。また、マニュアルの最後には表記の手引きおよび、卒業論文での歴代の誤字集（なかなか興味深い）を載せている。

実験の一つのテーマを週二回で行い、合計25テーマほどを実験する。各テーマにレポートが課せられる。学生には重荷のようである。テーマは年度ごとに少し変化するが、ほぼ以下を含む。

共振回路の応答、トランジスタの増幅、幾何光学、デジタル回路、電磁カウンタ組立、熱と半導体、トランジスタの特性、電子部品の統計測定、伝送線、交流ブリッジ、パルストランスの特性、写真实習、マルチバイブレータ、TTL電気特性、剛体振子、電位測定の実験、波形合成、透視変換、仮説検定、磁気ヒステリシス、機械振動、手回し計算機、弦の振動、発振回路、光の回折、慣性モーメント

以上で理解されるように、計算機の物質的なバックグラウンドがカバーされている。同時に手回し計算機などの計算機の仕組みを理解するテーマ、電位測定などの現物とシミュレーションの比較、電子部品の統計的な処理など、CSに意味があるものを組み込んでいる。またロボットや画像につながるテーマもある。

前述のものと同様に、現在では2年生の実験もわれわれ（CSコース）独自のものでなく、電子情報工学科全体で実施する体制になっている。したがって上記のものはこの形では実施できなくなった。これを前述のように、全体化や3年持ち上げで対処している。

3.4 3年生の演習

3年生の演習は作業としてはプログラムを作ることが中心である。やはり前述のようにプログラミング自身を教えることはない。一般に、教官は問題の意味を説明する。計算機端末室に学生を決まった時間に集めてキーボードを打たせることはしない。教官が問題を説明したあとはどうするかは学生の自由である。

全体としてかなり複雑なことが学生に求められる。しかしプログラミングに没入していく過程で、対象からのフィードバックに対応して考えるため、自然に目標に到達できる。科目内容には以下がある。

数値計算の特性、算法の評価などの演習：数値の性質やアルゴリズムの効率を具体的プログラムで調べる。学生には熟慮の上のプログラミングが求められる。いい加減なレポートは何度も再提出となる。

システムの設計と実現の演習：OS、計算機言語、知識処理、数理計画法など、トピックス的なテーマの集合である。カリキュラムからもれた部分を補う意味もある。

マイクロコンピュータによる実時間プログラミングの演習：実世界を計算機であつかう。物理的な実世界は並行して動いていて、計算機側はそれに即応した動作が必要である。学生はそのプログラムを書く。

3.5 3年生前期の実験

当学科の3年生の実験はソフト系とハード系に2等分される。前期（夏学期）には、各学生は、ソフト実験とハード実験とを約6週ずつ行う。学生は二つのグループに分かれ、ソフト実験とハード実験を行う。6週が過ぎて終了すると、両実験を入れ換えて行う。つまりある時点でみると常に両実験が同時に実施されており、各学生は一方を行っている。2年生の実験は実習としての性格が強いものに対して、3年生では次第に設計・製作としての側面が強くなるのが理解されよう。

3年前期ソフト実験

本実験は毛色が変わっていて、通常の実験とは異なり、教えるべき特定の学問内容があるわけではない。思考し、調査し、研究していく方法を教育する。実験テーマは「KJ法」、「調査研究」、「綴り方」という三つがあり、各2週(4回)をかけて実施する。

「KJ法」は、目的を実現するための思考過程に関する方法論として有名である。このテーマでは数人の学生ごとに、まず、自分たちで自由に具体テーマを設定する(たとえば「遅刻しないためにはどうすればよいか」)。そしてブレインストーミングから始まり、KJ法の手法に従い、目的達成に関して検討した諸意見を総合的にまとめる。議論は深夜に及ぶこともしばしばあり、つきあうのは大変である。最後に発表会を行う。このテーマの意義は、思考の方式を修得するというそれ自身の意義のほかに、自発性や自己内省の修得など、メンタルな意義も認められる。学生はその後自分の研究活動に役立てている。

「調査研究」もきまった対象があるわけではない。企業などでプロジェクトを起こすときには必ず準備として関連する諸問題を調査する。それを調査報告としてまとめ、それに従って意志決定をする。こうした仕事を目的とする。方法の中心となるのは「実地の活動」である。つまり企業、官庁、展覧会などに直接でかけて行って調査する。担当教官は数個のテーマを提示して、3~4人の学生がその一つを選ぶことから始まる。過去のテーマ例を以下に示す。

計算機言語の比較、LAN、ゲームソフトの歴史、マイコンの歴史、半導体市場の動向、自動演奏、仮名漢字変換機能の比較、個人ワープロの性能評価、音楽のデータ構造、OCRの製品と手法、音声認識、地理情報処理、地方自治体の情報処理、光ディスク、パソコンのOS、パソコン通信の現状、光情報処理、機械翻訳の程度、小中学校の計算機教育、卓上出版

提示するテーマを考えて学生に提供することは大切である。学生に目標をイメージしやすいもの、調査自体に今日的な意義があるもの、実地の調査で興味深いことが発見されそうなものなどが望ましい。学生には楽なテーマを選ぼうというものが数人いる。興味がある方向へテーマ選択をする

よう勤める。楽なテーマでは高度なことを求める、と釘をさしておく。

学生は、テーマを掘り下げ、詳細な調査項目を設定し、分担を決め、報告書をまとめ、最後に発表会を開催する。発表会は一般の会議のプレゼンテーションの形式をとる。つまりOHPを使い、時間内に質疑応答を行う。これが初めての公式な形の発表である。発表に関しては、「OHPの字を大きく」、「発表の時間を正確に」などの基本的な注意事項は指示する。最近の学生の気質として「うけ」重視がある。「うけをねらえ」と提言すると驚くほど上手に発表するグループがある。

「綴り方」は手紙の書き方など、文章表現の練習を行う。大学でここまでしなくても、という考えがあろう。しかし、文章の形式を把握した上で、文書を作成することは確実なコミュニケーションには必要である。

3年前期ハード実験

2年生の実験が「測定」が中心であったのに比べ、3年前期のハード実験では、測定を多く含むものの、回路の製作という観点も強まる。これは次第にanalysisからsynthesisに近づくことを意味する。ソフト実験と同様、学生は6週にわたって(2週×3テーマ)実験する。テーマは以下から3テーマが学生に割り当てられる。約3人が一組になり、一つの実験を行う。

定電圧電源回路、演算増幅器とアナログ演算、D/A・A/D変換、マイクロプロセッサ、直流サーボモータの特性と制御、半導体センサ

3.6 3年次後期の実験

後期(冬学期)の実験も前期と同じ割り振りでソフト実験とハード実験とを半分ずつ行う。これらはシステムの製作という側面がさらに強く、synthesisの程度が強い。前期ではソフト、ハード各3テーマずつ行ったが、後期では各1テーマずつである。ここで修得したことは卒業論文研究で生きてくる。この実験は卒業論文のミニ版、あるいは予行練習であるといえる。一人の学生がソフトとハードの両テーマを行うことは、理念で述べた「CSのバランス」の考えによる。

3年後期ソフト実験

3年後期のソフト実験は、約6週で一つのテーマを実験する。毎回なるべく異なるテーマを考え出す。学生2人に一テーマを与える。以前は学生

4人程度を組にしていた。しかし、能力の高い学生が独走する弊害があったので現在は2人ずつである。2人のときも分担し、協力するように指導している。テーマは黒板に書いて説明したあと、学生が選択する。選択させることは学生の能動性を自覚めさせるために重要である。

担当はソフト系の若手教官全員で行う。教官当たり2、3テーマを毎回担当する。学生が興味をもつテーマを準備する。そのためには教官自身もおもしろいと思うものがよい。テーマをどう料理するかは学生の自主性にまかされる。もちろん放任ではなく、意味のある方向に誘導することがある。「ゲームプログラム」というテーマで行った実験の例を述べる。学生たちにその中で知識処理をさせようとしたが、学生側からは、「テトリス」(落下図形をはめ込む、有名なパズルゲーム)を作りたいという申し出があった。そのままでは単にコーディングの問題になるおそれがあったので、教官側で、「テトリスをプレーするプログラム」(人間にプレーさせるのではない)を作ること勧めた。これをするには、人間の行動を観察して知能的なプログラム動作として実現しなければならない。これは人工知能の基本的発想として意味がある。学生は、プログラムのテトリスの上手さを観察するだけで、自分の設計の良否を認識できる。結果としてこのアイデアは成功して、学生たちは意気高く発表会で実演することとなった。今までの実施テーマ例を示す。

中学1年の機械翻訳、数字認識系、多倍長数系、データベース(音楽、図書、学科内知識など)、仮名漢字変換、文字種統計収集系、電子掲示板、郵便料金エキスパート系、状態遷移図表示系、ゲーム系(五目ならべ、HEXなど)、構文解析系、ミニ言語処理系(Pascal, Prolog, Lispなど)、プログラム動作解析系(C, Lisp, FORTRANなど)、数式処理系、チューリング機械シミュレータ

問題点としては、計算機言語の学習の問題がある。未知の言語に挑戦することを少しいやがる学生もいる。ワークステーションや大型計算機を使わず、パソコンで簡単に済ませようとするものもいる。これに対してはテーマに向けた機器を選択するように誘導する。

3年後期ハード実験

ソフト実験と同様に、ハード実験も6週にわたって学生は一つのテーマに取り組む。教官が回路図を与えることはなく、テーマの概要、方式に関する説明、目標とする仕様を学生に与える。学生は4~5人でチームを組み、自分たちで「もの」を設計し製作する。苦勞すれば完成するテーマを与えるが、ときに失敗しそうな組もでる。そのときでも教官は誘導するだけである。学生の達成感が重要だからである。学生に対して毎回異なったテーマを与える。学生に初回のものでも、教官側に経験があると、おもしろさを阻害する。テーマ設定の準備は、実施途中の時間に加え、苦勞が大きい。この実験は当学科の指向するパワーエデュケーションの典型である。今日までのテーマ例を示す。

数字認識器、リニアステッピングモータ、文書読み上げ機、ポインティングガン、音響風速計、発表会用タイマ、自動焦点合せ器、ロボットアーム、デジタル音声録音器、メモリスコープ

このようにテーマは一定の明確な利用目的をもち、機能は必ずデジタル回路を設計して実現する。

3年後期の実験は最後に発表会を行う。ソフト実験とハード実験とを合同して行う。両実験は学期の中間で入れ替わるので、発表会は年に2回ある。全部で10~15テーマあり、午後いっぱい費やす。教官は特に用のない場合は出席している。この発表会では、OHPによる通常のプレゼンテーションに加え、実演を行う。

ハード実験の実演では製作した機器を壇上で動作させる。ソフト実験の実演では、会場となる教室にある三つの大きいモニタに製作物の実行を表示する。発表会はなかなか楽しいものである。学生も自分の成果を見せようといきり込んでいる。ときに動作の異常なハードや、バグのあるソフトがあり爆笑となる。徹夜で回路を完成させても、実験室から教室に移動すると故障することがある。ソフトを直前に改良しようとしていじると本番で往々にして動かなくなる。そうした経験にも意味がある。発表会は盛り上がりつつ終了する。

3.7 3年生の輪講

3年生を各教官に割り当て、10週程度の間、輪講形式で英語の講読を行う。これを前期・後期に

一度ずつ行う（電子情報工学科 CS では前期だけ、6週間×2回に短縮）。各教官に対して6名ほどの学生を割り付ける。したがって密度が濃い。一人の学生にはソフト系の教官とハード系の教官両方が割り当たる。講読する読み物は教官ごとに異なる。CS の基本的な資料の類が多い。学生は、(日本の英語教育のせい)単に英語を日本語に訳せばいいと思っているのが常である。この輪講を通じて学生は日本語に訳すだけでは通らないことを悟る。極端に言えば日本語訳は不要である。いったん理解した内容を説明する感覚を多くの学生は分かってくる。この輪講で、工学やCS の基本用語の語義を身につける。一例として、implement, default, unique, explicit, available などの言葉は、基本的にも関わらず、多くの学生は10年近い英語教育で身につけていない。また辞書を引いても簡単には分からない(上の例では、学生が「実行する、不履行、ユニークな、明確な、役に立つ」などと訳するのが典型)。この輪講ではこうして、実資料を読むことにより、技術の分野での実用の英語を修得する。

3.8 4年生の輪講

これは卒業論文に関連する科目である。3年の輪講で修得した英語の資料を読む力を使って、学生は、卒業論文研究の文献を読む。一まとまりの論文を一人で明確に理解する訓練はこの段階でもまだ必要である。

発表するものには、数ページの短い論文や解説文が多く、卒業論文の指導教官と本人とが研究室で決める。現在では二つの会場で並行して行う形に落ちついている。一つの会場で約5人ずつ発表し、一人の教官が付き発表にコメントを与える。ほかに1, 2人の教官が参加していることもある。学生には出席の義務があるが、どちらの会場にいてもよい。

学生は一人あたり、20分程度の発表を2回する。学生から見ると重い負担である。これで発表が洗練されてくる。学生はすでに調査研究、実験(2回)を含め3回以上の発表を経験しているが、共同であり、演壇で声を出すのが未経験の学生も残っている。この発表で全員が自身による発表を経験することになる。

3.9 学生の研究室活動

学生は、必須科目の実験や演習を終了し、規定の単位数を取得していると、講座(あるいは研究室)に配属され、卒業論文を目指す研究をする。講座とは研究の基本単位で、教官側は教授、助教授、助手、技官(一部欠けていることが多い)からなる。研究室はその半分の人員である。単位を落とさず、卒業論文を始める学生が実質的な4年生である。

3月の初めに、実験や演習などの教官が共同して担当している科目の成績判定会議が開かれる。その直後に新4年生の講座(研究室)への配属を決める会議が開かれる(ドラフト会議と称している)。会議に先だって、卒業論文のテーマ説明資料を学生に向けて掲示する。これを見て学生は配属志望を提出する。若手の教官が能力のある学生を獲得すべく努力するため、テーマ説明は長く懇切丁寧なものになる。

現在のところ志望の過度の集中は起きていない。現在の配属の方法は、学生の志望を尊重することと、2, 3割の増減を許容して各研究室に分けることである。2~3年で平均してみると、学生は、適度にばらまかれている。研究室の人員数でも、ほぼ平均している。今までの経験では、学生の90%以上が第1志望の研究室に配属されている。学生に志望を出させ、その志望を満たしてやることは士気の面で重要である。

4. ま と め

東京農工大学工学部(旧)数理情報工学科、現在電子情報工学科 CS コースの専門教育について理念、方法、特色などを述べた。その理念は真のCS を対象とし、その特色は創造性、実際性を指向し、手作りで設計し実施してきた点にある。ただし、それにより実現されたものは多様な色合いをもつ。この理念、方法、特色は、本文にあるように、日々の議論によって毎年少しずつ変わっている。電子情報工学科以降のカリキュラムが過密化したことに対し、現在は対症療法的な処理をしている。これを解決し、独立したCS のカリキュラムを実現するめどをたてつつある。上述の考えや手法などは、本音を吐露した部分が多い。是非ご意見をお聞かせ願えれば幸いである。最後に、本文の内容は、当学科教官全員の日常的な活動の

なかで生まれたものであることを、付け加える。
なお、具体的なカリキュラムの詳細は文献 1)~5)などを、また特色のもとになった考えについては 6)などを参照されたい。

参考文献

- 1) 野口正一, 中森眞理雄: 大学等における情報処理教育の諸問題, 情報処理, Vol. 31, No. 10, pp. 1379-1389 (1990).
- 2) 東京農工大学工学部数理情報工学科十年史, 数理情報工学科十周年記念委員会 (1986).
- 3) 小谷善行, 阿刀田央一, 中森眞理雄, 高橋延匡: 情報工学系学科における実験・演習の一設計例, 情報処理学会論文誌, Vol. 22, No. 5, pp. 402-410 (1981).
- 4) 高橋延匡, 阿刀田央一, 小谷善行, 中森眞理雄: 工学系学科における実験・演習 I, bit, Vol. 12, No. 15, pp. 2010-2013 (1980).
- 5) 高橋延匡: 各大学情報工学科の紹介—東京農工大学工学部数理情報工学科, bit, Vol. 12, No. 15, pp. 2074-2075 (1980).
- 6) 高橋延匡: メタ情報工学の試み, 情報処理学会プログラミングシンポジウム・夏のシンポジウム「情報処理教育」報告集 (1991).
(平成3年8月2日受付)



小谷 善行 (正会員)

昭和 24 年生。昭和 46 年東京大学工学部計数工学科卒業。昭和 52 年同大学院博士課程工学系研究科修了。同年東京農工大学工学部数理情報工学科講師。現在同大学電子情報工学科(コンピュータサイエンス)助教授。記号処理言語を含むソフトウェア工学および知識処理に興味を持つ。人工知能学会, 日本ソフトウェア科学会, 電子情報通信学会, 認知科学会各会員。



高橋 延匡 (正会員)

昭和 8 年生。昭和 32 年早稲田大学第一理工学部数学卒業。同年(株)日立製作所中央研究所入社。HITAC 5020 モニタ, TSS の開発に従事。昭和 52 年より東京農工大学工学部数理情報教授。平成元年電子情報教授。理学博士。オペレーティングシステム, 日本語情報処理, パターン認識の研究に従事。電子情報通信学会, ソフトウェア科学会, 計量国語学会, ACM 各会員。

