これからのオブジェクト指向設計に必要な形式論理の訓練の実践例
― ミニ・プルーフ・チェッカMizar-MSEを用いて ―

職業能力開発総合大学校東京校　　福　良　博　史

An example of practical training with Mizar-MSE for the formal logic that is base of object-oriented design
― Education and training that uses the mini proof checker ―

Hirofumi FUKURA

Summary

The purpose of "education in mathematical logic" is that the students come to understand the concept of formal methods by use of OCL. Mizar-MSE is an effective tool for the students that learn the logic about propositional logic and predicate logic. Trial education and training using the tool was performed to the sophomores of the information technology course. It found that the way is useful in understanding of the formal methods.

1．はじめに

ソフトウェアの信頼性を求める機運が数年前[1)]から高まってきている。とくに組込み系のシステムにおいては人命にかかわるようなシステムが多数存在する。たとえば、宇宙ロケット、飛行機、車、エレベータ、医療器具などが例として挙げられる。このようなシステムを開発し、サービスを提供する上で初めに「このシステムは人命を保護できているか？」ということを考えなければならない。

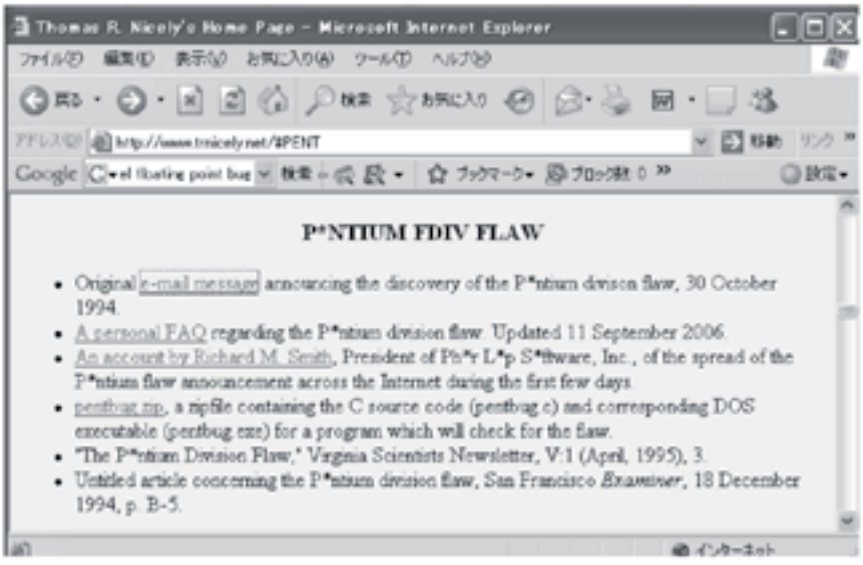


図１　浮動小数点演算のバグ報告（発見者のホームページ）

ソフトウェアの設計・実装において、テストを１万回行ったとか、徹夜でテストをしたとかいったところで努力は買っても信頼性の保障にはならない。たとえばテストしなければいけない条件分類が正しくできており、その確認テストがなされているか、その保障をどのように得ているのかを正しく評価できて初めて信頼性の議論ができる。消費税の計算を、単価に1.05を掛けて求めるような場合を考えてみる。単価が1,000円の時の検査が確認できていれば、100円の時の検査が省略できるであろうか？という議論もある。通常ここまで考える必要は無いだろうと思うかもしれない。しかし、1994年10月にThomas Nicely教授がIntelのPentiumプロセッサの浮動小数点演算にバグがあることを報告（図１）した[2)]。このようなバグの可能性があるということは、極論すると、１＋２の結果が３となることを確認したが、３＋４の検査をしていなければその演算結果の値が７となる保障が無いかもしれない、という議論に発展しかねない。このような検査を100％実行することは非現実的かつ実際のものつくりの現場では耐えられない。それではこのような問題に対処するためには、どのような方法を考えればよいのか。現在の技術でこのような問題に対処するための方法としては、形式手法（Formal Methods）と呼ばれている方法[3)4)]が考えられる。この手法の大きな特徴は、数学の形式的な証明の方法論を、ソフトウェアの設計段階、検査段階などで利用し、信頼性を高めることにある。図２には、形式手法の紹介のホームページの一部を抜粋して示す。この55にはMizarが、61にはOCL（Object Constraint Language）が紹介されている。

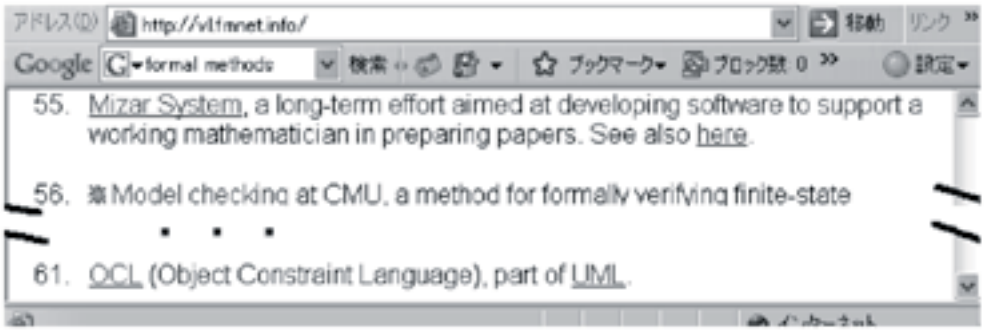


図２　Formal Methodsの方法論が紹介されているホームページ（抜粋）

この形式手法は上記のホームページを見ても判るとおり、沢山の方法論がある。しかし一般的にはどの方法論も通常のプログラミング言語を覚えるような容易さは無い。高度な教育と訓練が欠かせないのが実情と考えられる。このような中で、現在一番啓蒙活動が盛んなオブジェクト指向による設計方法論のUMLの一部としてOCLという言語がある。このOCL言語は、形式手法に則った記述方式を採用している。このために、上記ホームページにも紹介されている。このOCLは、証明の推論を行うことが目的ではなく、システムの或るポジションやタイミングにおける、あるべき姿を形式的に条件記述によって定義するための言語としての位置づけのため、利用の仕方は比較的容易な言語と考えられる。しかし、このOCL自身は記号論理の素養が無ければ、誤った解釈をする恐れがある。翻訳されているUMLのテキストのなかには、記号論理の素養が無いためか、たとえばimplyを「暗示する」と誤訳しているものが見受けられる。このOCL言語は、これからのソフトウェア技術者にとって必須の道具となる可能性があり、この言語を理解するためには記号論理の教育訓練が必須となるため、昨年度はこの教育についての提案を行った[5)]。今年度は、情報技術科（東京校の短大）の２年生に短期間ではあるが、半日９回

の実習（2単位分）を試行することになり、その有効性が確認できたのでここにその結果を紹介する。

2．実験的に行った記号論理教育の教科細目

この教科の目的は、OCL言語を理解するための素地をつくることにある。このOCL言語は、記号論理の考え方を採用している。この言語を実装したツールUSEがあるのでUSEの利用方法を紹介し、どのようなことがOCL言語で記述できるのかのイメージを持ってもらう。その上で、集合論、真理値などを学習し、記号論理の実習を行う。OCL言語は、仕様記述を支援するものであるため、プルーフ・チェッカーではない。ツールのUSEは、OCLで定義した内容と、クラスから生成されたインスタンスの整合性チェックを行うのが目的のため、プルーフ・チェッカではない。つまりUSEは記号論理を理解していないと使えないが、記号論理の教育用ツールとしては不向きである。Mizar-MSEは、記号論理の入門用のプルーフ・チェッカーとして作られているので、記号論理の実習には、Mizar-MSEを用いることとした。

昨年度、記号論理の教育案[5]を提示した。今回はその半分程度の内容について試行した。おおよそ表1の通りの内容にて試行した。

表1　試行の概要

日	教　科　細　目
1日目	USE（UML用ツール）の紹介
2日目	集合論
3日目	関係
4日目	真理値表
5日目	Mizar-MSEを用いた命題論理実習1
6日目	Mizar-MSEを用いた命題論理実習2
7日目	Mizar-MSEを用いた述語論理1
8日目	Mizar-MSEを用いた述語論理2
9日目	全体の復習

3．具体的な指導方法

初日は学生にUSE[6]というツール（図3）の使い方の説明と実習を行った。このツールは、Javaで作られたフリーソフトでUMLのOCL記述のモデル検証ができる。

図3の右上は、クラス図、右下は、オブジェクト図が表示されている。左下のcontextで始まる文がOCL言語によって書かれている。この図のオブジェクト関係における各種関係がある中で、この文は一つの制約条件を記述している。この文の中に"forAll"や"implies"と書かれているのが読み取れる。これらのキーワードが記号論理の用語でもある。

OCLの記述方法とその検証方法を学生に説明し、実際に動かしてもらう。そして記号論理の必要性を把握しておいてもらう。このツールを使うことは、これからの記号論理の必要性のイメージを体験してもらうことにある。何人かの学生は非常に興味を持って取り組んでくれていた。

2日目から4日目までは、教師が基本を説明し、学生は交替でホワイトボードのところに出てきて、演習の課題に取り組み、皆で板書の結果の正誤を確認をするようにした。このことによって学生全員が、何が間違いなのか、何が正しいのかを理解しあえた、と考えている。

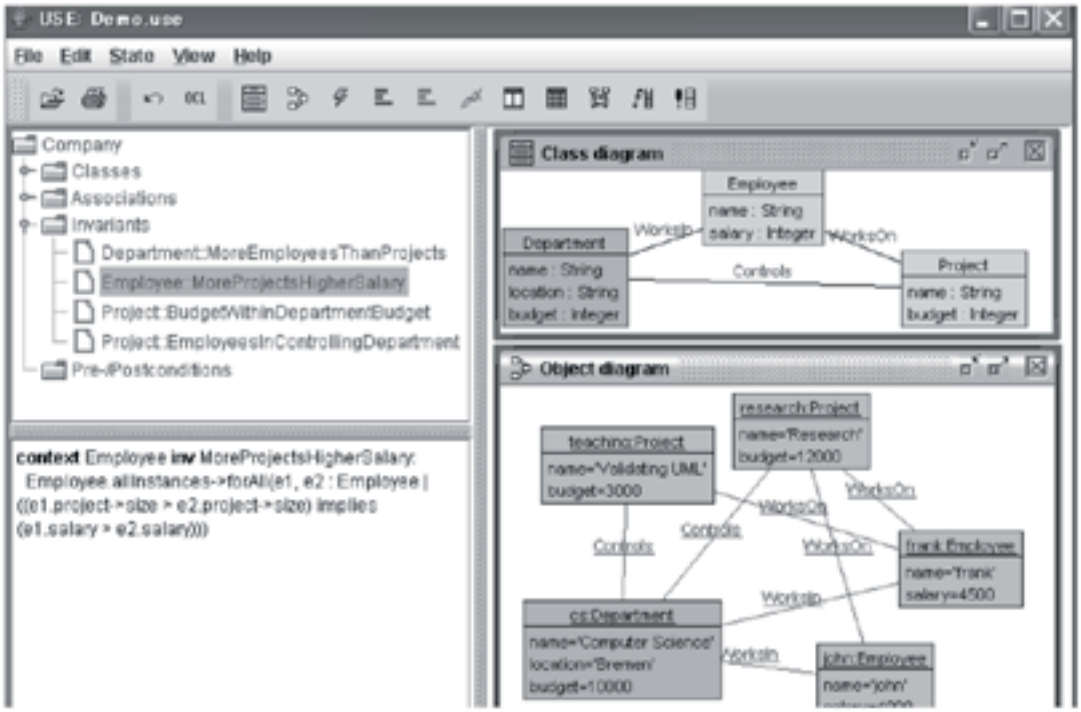


図 3　USEの利用例

5 日目以降は、Mizar-MSE[7] という記号論理の簡単なツールを用いて実習を行った。なお、証明が正しくないか、構文にミスがある場合は、図 4 の"Thanks"と表示されている箇所に"SORRY"と表示される。"Thanks"という表示は、推論に間違いが無かったことを示している。

図 4　Mizar-MSEの実行例

このMizar-MSEというツールは、海外において命題論理と述語論理の学習用に利用されている。baby Mizarとも呼ばれている。このツールは、現在カナダのAlberta大学においてマニュアル、ツールのソースコード等が公開されている。インターネットにつながったパソコンからは、直接ネット上でこの大学が管理しているツールがweb上から使えるようになっている。このツールを選んだ理由は、このツールのソースコードが公開されていることと、GUIが無くコマンドライン上での操作ではあるが、言語仕様が割合シンプルなので初心者からでも取り組み易いと判断したことによる。今回の実習ではweb上のツールを用いず、わざわざ当校においてツールを作り変えた理由は、何らかの原因によりインターネットとの接続に不具合が生じると実習ができなくなるおそれがある。このリスクを回避するためには、なるべくスタンドアロンの環境で使えるツールが実習には望ましいとの判断による。そして、副次的な効果ではあるがMizar-MSEが子供版とすればその大人版ともいえるMizar[8] との親和性に富むため、Mizar-MSEでは不都合な場合は、Mizarを使う必要が生じる可能性があり、そのための第一ステップも兼ねられると考えた。

このツールを使うために、以下の 3 点を改造し、東京校バージョンとして利用することとした。

(1) もとのツールは、UNIX用のため改行コードとして"LF"のみを評価していた。このためWindowsでも使えるように"CR+LF"をも改行コードとして評価するように変更した。

(2) コメントが、"=="のあとから改行コードまでがコメントとなっていた。シフトキーを押しながらでないと"="キーは使えないので不便であるし、C++やJavaに慣れ親しんでいる学生には、コメントは"//"を用いるほうが使いやすいと考え元の"=="以外に"//"もコメント用の指示とするように改造した。

(3) Mizar-MSEの述語論理の全称限量子（universal quantification）は、"for"というキーワードになっている。これはMizarと同じキーワードを用いている。これを"forAll"とした。理由は、通常の文献では、"∀"という記号を用いるかまたはテキストのスタイルとして"for all"と表現する場合が多い。その上当校での実習の趣旨は、OCLの理解促進を目標としていることを考えると、OCLでは"forAll"を用いていることから、当校で利用するMizar-MSEも"forAll"を用いることが妥当と判断した。また学生も単なる"for"より"forAll"となって"All"という言葉がついているほうが全体という意味合いが明確になると考えた。

以上の改造をWindowsのCygwin環境で行い、東京校版のMizar-MSEを作成し、学生に配布し使ってもらうこととした。

4．Mizar-MSEの使用例

4.1　Mizar-MSEの主な表記法

Mizar-MSEは、一階述語論理を扱う。以下の表 2 にMizar-MSEの主な記法を示す。参考までに日本で出版されている記号論理関連のテキスト[9)10)] で用いられている一般的記法を併記する。

表中のNo. 1 〜 9 は、一般の記号論理の記法なので、それほど違和感は無い。No.10は、証明が矛盾している場合、言葉で矛盾している旨を書くことになる。しかしMizar-MSEはコンピュータで処理する言語であるため、自然言語の表記では困る。そこで、矛盾している場合には、contradictionというキーワードを矛盾した箇所に明記するようになっている。No.11は、ソートの定義用の記法である。Mizar-MSEのMSEとはMulti Sorted Environmentの略で多ソート論理[11)] を扱える。このため、given Socrates being Humanと書くとSocratesはHumanというソート（型）であることを宣言している。

ソートは、はじめから決められた（いわゆるprimitiveな）ものは無く、自分でそのつど必要なソートを定義する（例えば、given i being INTEGER等）ようになっている。なお関数は扱えない。たとえば、述語でAdd（1，2，3）を"1と2の和は3である"、と表現することはできる。しかし、Add（1，2）が関数として値を持つような表記はできない（注：Mizar-MSEではなく、Mizarは関数表記の機能を持っている）。

表2　Mizar-MSEの主な表記法

No.	一般的記法	意　　味	Mizar-MSEの記法
1	¬	否定（negation），not	not
2	∧	連言（conjunction），and	&
3	∨	選言（disjunction），or	or
4	⇒	含意（implication），imply,if then	implies
5	⇔	同値（equivalence），if and only if	iff
6	∀	全称限量子（universal quantifier），for all	forAll
7	∃	存在限量子（existential quantifier），exist	ex
8	ϕ	命題　（例）ϕ：ソクラテスは人である	P[]
9	ϕ（x,y,…）	述語　（例）ϕ（x）：xはϕである	P[x,y,…]
10	なし	矛盾（contradiction）	contradiction
11	なし	ソート定義　（例）x　integer	given α being β

Mizar-MSEの文の構成は、おおよそ表3のとおり。

表3　Mizar-MSEの推論の構成概要

```
environ
        ・・・          //":"の左は参照用のidラベルの定義
    αi：  xxxx；      //前提条件列挙
    αi+1：xxxx；      //（ソート定義、公理、前提となる定理等）

        ・・・
begin
        ・・・
    βj：xxxx；          // 推論の列挙
    βj+i：xxxx    by αm, βn；
                        //  by  以下に、この行の推論の根拠となる行のidを記す
        ・・・
```

注：(1) 文の終わりの表示は、"；"を使う
　　(2) コメントの開始は、"=="または"//"で始める

4.2　命題論理の例（化学反応への応用例）

この例は、以下の化学反応についての証明を行う。

（1）$MgO+H_2 \rightarrow Mg+H_2O$

（2）$C+O_2 \rightarrow CO_2$

（3）$CO_2+H_2O \rightarrow H_2CO_3$

以上のことを前提とした場合に、

MgO、H_2、C、およびO_2があれば、

H_2CO_3ができることをの証明を表 4 に記す。

表 4　命題論理の例

```
environ
P0: MgO[] & H2[] implies Mg[] & H2O[];
P1: C[] & O2[] implies CO2[];
P3: CO2[] & H2O[] implies H2CO3[];
begin
100: now
  assume 20: MgO[] & H2[] & C[] & O2[];
  1:  MgO[] by 20;
  2:  H2[]  by 20;
  3:  MgO[] & H2[] by 1, 2;
      Mg[] & H2O[] by P0, 3;
  11: Mg[] & H2O[] by P0, 1, 2;
  12: H2O[] by 11;
  3:  C[] by 20;
  4:  O2[] by 20;
  21: CO2[] by P1, 3, 4;
  thus H2CO3[] by P3, 21, 12;
end;
MgO[] & H2[] & C[] & O2[] implies H2CO3[] by 100;
```

4.3　述語論理の例

この例は、

（1）人は皆死ぬ

（2）ソクラテスは人である

表 5　述語論理の例

```
environ
reserve x for Human;
given Socrates being Human;
A2: forAll x st man[x] holds mortal[x];
A3: man[Socrates];
 begin
mortal[Socrates] by A2,A3;
```

以上のことを前提とした場合に、

「ソクラテスは死ぬ」という推論が正しいことを述語論理の記法で証明する例を表 5 に示す。

5．学生へのアンケート結果

アンケートは無記名で、多肢選択形式を基本とし、コメントの自由記述も可能な形式とした。回答は、20人の内、当日２人欠席し、１人未記入のため、17人から得られた。表６に集計結果を示す。

まず、今回の実習に対する学生の観点からの必要度合いについて確認する。これは、質問１（UMLのツール），質問２（集合論），質問３（記号論理），質問４（記号論理のツール），質問５（教材）という切り口で、不要か否かを選択肢に加えておいた。このうち、不要という選択をした学生は、集合論に１名いた。それ以外の項目については、不要と答える学生はいなかった。

集合論について、難しいと答えた学生が四分の一いる。しかし逆にわかりきったことだから不要と考えている学生が１名いたことになる。なぜわかりきったことと判断できるかというと、集合論が不要と答えた学生は、他の質問もほとんどが易しいと回答している。ツールのUSEに対してだけ、もっと詳しくと記している。

興味深い反応が見られるのが、ツールに対しての態度で、USE（46%）についても、Mizar-MSE（34%）についてももっと詳しく、実習をして欲しいという欲求が多い。それに反し、座学の集合論（12%）と記号論理（18%）については、もっと詳しくという欲求は、ツールを使う場合と比較してかなり低い。

ツールについて難しいという比率は、USE（24%）もMizar-MSE（24%）もどちらも同じとなっている。しかしこの内容は以下のような差異があると考えられる。USEについては、１日だけの訓練時間しかとっていなかった。しかも記号論理の基礎が無い状態で使って、動かしただけなので理解は不十分と学生自身が認識していると考えられる。これは、ツールについて易しいと回答した割合がUSE（６%）とMizar-MSE（24%）でUSEのほうが四分の一に留まっていることから類推できる。Mizar-MSEについては、記述式のコメントで、エラーの切り分けが難しいと答えた学生がいた。Mizar-MSEは、最近のコンパイラのようなきめ細かな、行単位でのエラーの指摘をしてくれないので、理解しづらいという面から、難しいという答えが返ってきていると考える。そのような使い勝手の悪いツールではあるが、Mizar-MSEは、易しい（24%）と答えた学生の比率が高い。

6．まとめ

この実習中に数人の学生は、Mizar-MSEでの証明の課題に非常に熱心に取り組んでいた。その学生達と話し合った結果判明したことは、(1) 教師にいちいち細かく色々聞くのは気が引ける、(2) 座学の場合には、学生同士で話し合っても理解できないことは、教師に聞く以外に方法が無い、(3) ツールを使う場合は、ツール自身が詳細なチューターの代替となってくれる、(4) ツールの場合は、気兼ねせずに、色々細かいことまでツールに尋ねることが平気で出来る（自分で色々試して見ることにより、その結果をツールが良いか悪いか判断してくれる)。このことは、アンケート結果の、座学よりもツールを用いた実習に対してもっと詳しく実習をして欲しいという積極的な欲求が強いことからも窺える。

以上のことから、記号論理の教育に座学ではなく、自分の推論を検証してくれるツールを導入することは、個人の能力に合わせた進度での個別学習的な対応が可能となり、個人別によりきめ細かく、より厳密な論理の進め方を訓練することが可能と判断する。つまり一種のe-ラーニングとなっている。しかもこのツール自身はフリーでLinuxでもWindowsでも操作可能なため、パソコンを持っていれば自宅で何時でも自分の考えを、厳密に検証することができる。

今回の実習を通して、ツールを用いた記号論理の教育訓練に学生は興味を持ってくれたと考える。し

かし、本来の信頼性の高いソフトウェアの設計を通した「ものつくり」のための基礎となる教育を目指すためには、以下の点について整備していく必要があると考える。

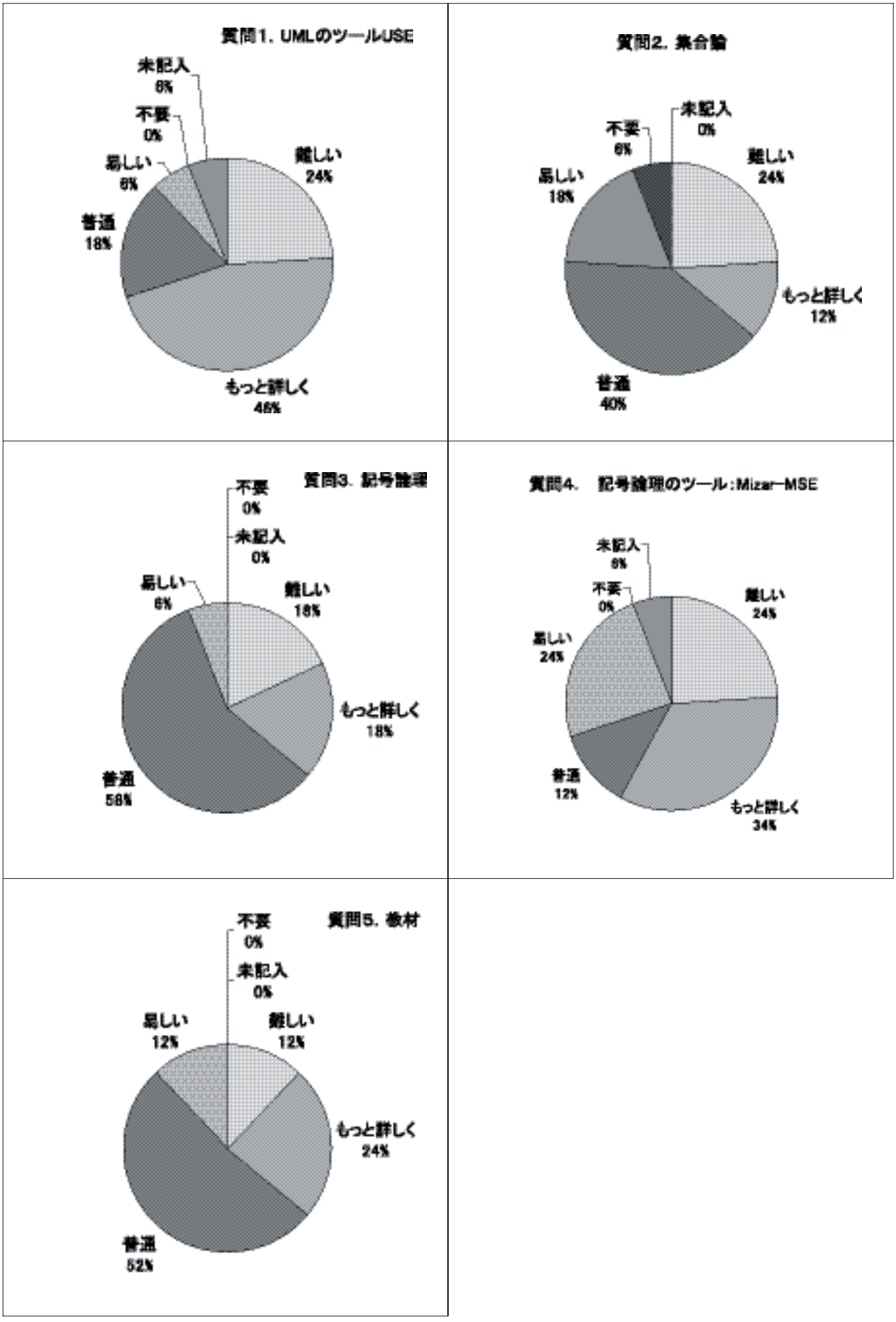
（1）Mizar-MSEの構文および推論のエラーチェックをきめ細かくできるように改造する。
（2）Mizar-MSEの命題・述語等に日本語が扱えるように改造する。
（3）実習時間数と実習内容を現場の実情に合わせて 2 単位版または 4 単位版などのいくつかの系統を用意する必要がある。
（4）現在実施されているソフトウェア工学実習等におけるUMLによる設計方法論との連携・整合性に配慮することが無駄を省く上で必要となる。
（5）状態遷移図等、記号論理の応用例題用教材を開発し、形式的な仕様記述の方法に慣れておくような訓練を追加し、より信頼性の高い設計方法のありかたを実感として理解できるような工夫を検討する。

この（5）の教材開発の進み具合によっては、組み込み系のシステム開発への応用教材としても有効となる可能性がある。組み込み系でも使える可能性があれば、情報系のみならず電子系の学生も実習の対象と考えていく必要があるかもしれない。

〈参考文献〉

1）http://www.ipa.go.jp/SPC/report/02fy-pro/html/product.htm
2）http://www.trnicely.net/#PENT
3）http://vl.fmnet.info/
4）福良博史："ソフトウェア技術者の道具としての「formal methods」"、茨城職業能力開発短期大学校　紀要、1998、pp.17〜22
5）福良博史："情報技術に必要な記号論理の教育 ―UMLその他の現在の設計手法との関係―"、職業能力開発報文誌、Vol.18　No.2（2006）、pp.65〜70
6）http://www.db.informatik.uni-bremen.de/projects/USE/
7）http://www.cs.ualberta.ca/~piotr/Mizar-MSE/index.html
8）http://mizar.org/
9）廣瀬健："情報数学"、コロナ社、1994、pp.8〜11 pp.262〜9
10）井田・浜名："計算モデル論入門 ―チューリング機械からラムダ計算へ―"、サイエンス社、2006、pp.23〜25
11）J.D.Monk："Mathematical Logic"、Springer-Verlag,1976,pp.483〜485

表５　アンケートの集計結果



本誌は当大学校教職員の主著又は共著（学外者含む）による職業能力開発に関する総合的研究論文誌です。

掲載する論文のカテゴリーは

① **論　文**：特定の主題に関する研究の成果を体系的に論述したもので、仮説の検証、理論の定立、その他独自の価値を主張しうる内容をふくむもの。

② **研究ノート**：調査の実施、先行研究の整理等の結果、新たな仮説或いは研究の方法論を提示したものなど一つの体系的研究の一部であるが、それ自体として一応完結し、引続き行われる研究の方向づけを与えるもの。

③ **資　料**：他所にないデータを整理、分析したもので、これを公にすることが研究及び職業能力開発関係者にとって有益と考えられるもの。

④ **紹介又は解説**：内外の職業能力開発界の動向、文献、その他注目すべき情報を体系的に説明したもの。

の 4 部門です。

職業能力開発研究　第25巻

発　　行　２００７年３月
編集・発行人　**職業能力開発総合大学校能力開発研究センター**
　　所長　　緒　方　　悟
　　〒229-1196　神奈川県相模原市橋本台4-1-1
　　TEL 042-763-9155（普及促進室）
印　　刷　システム印刷株式会社

ISSN 0919-9594

HUMAN RESOURCES DEVELOPMENT RESEARCH

VOL. 25 2007

TEATISE

THE INSTITUTE OF RESEARCH AND DEVELOPMENT
POLYTECHNIC UNIVERSITY

4-1-1, Hashimotodai, Sagamihara, Kanagawa, Japan.