

論理プログラミングのための非手続き型言語の教育

芹沢照生

工学院大学 電子工学科

工学院大学では1992年度より非手続き型言語の教育を行っている。

Education of non-procedural language
for logic programming

Teruo Serizawa
Department of electronics, Kogakuin University

In Kogakuin University, a new curriculum was introduced in 1992 in order to teach non-procedural language.

1. はじめに

工学院大学電子工学科情報工学コースでは、1・2年生を対象としたプログラミング言語教育を全面的に見直すこととなり、1992年度から新しいカリキュラムによる教育を開始した。1年生では、手続型言語（Pascal）を用いて基本的なアルゴリズムを学び、2年生では、非手続型言語を用いて記号処理などの非数値処理を学ぶこととなった。（2年生の後半では、C言語を学ぶ）

今年で、2年目となった非手続型言語の教育の現状について紹介する。

2. 手続型言語と非手続型言語

Pascal、C、Fortran、BASICなどの言語を手続型言語という。これは、ノイマン型コンピュータのアーキテクチャーにもとずき計算機上での実行手順をほぼそのままに、人間にも解りやすい形で記述したものである。

これにたいし、問題を解く手順ではなく、問題にかかわる性質等を記述し問題を解こうとするプログラム言語が非手続型言語である。Prologは述語論理の論理式をプログラムとみなすことにより問題を解決する論理型の非手続型言語である。

我々は、このような従来の言語とは考え方の異なった言語を学ぶことにより、プログラミングにたいする柔軟な思考方法を身につけ、それぞれの問題の解決に適した言語を選択する能力を学ぶことを目標にカリキュラムを構成した。

3. Prologのプログラミング

Prologのプログラムは、事実、ルール、質問の3つの部分から出来ている。簡単な例を以下に示す。

事実：	ソクラテスは人間である。	human (ソクラテス) .
ルール：	人間は死ぬ。	mortal (X) :- human (X) .
質問：	ソクラテスは死ぬか？	?-mortal (ソクラテス) .

Prologのシステムはこの問いにたいして、事実をルールに適用して正しい（yes）と答える。これは三段論法の適用の一例である。このように述語論理のルールに従って、質問を証明すべき定理とみなし、既に与えられているデータを公理系として証明を試みるのがPrologの仕組みである。

この例のように真偽を直接問う方法のほかに、質問の中に変数を入れておき、その質問が正しくなる変数の値をすべて求めるという使い方も出来る。

4. Prologのカリキュラム

Prologの教育には、1人1台のパソコンを用意し、演習中心でおこなった。学生向きの手頃な教科書が見つからなかったので、参考書を指定するにとどめ、演習は毎回プリントを配ることで行った。

使用したシステムはPC9801とRUN/Prologである。処理系の選択には、1) 文法が標準的であること 2) パソコン上で実用的に使えること 3) 漢

字が使えること 4) 価格が安いことなどを主な理由とした。また、自宅のパソコンを使いたい学生のためにはフリーウェアの処理系の紹介もおこなった。

以下にプリントで用いた例題を示す。

「サザエさん一家」

親子関係を示すデータを使って、事実、ルール、質問の基本3要素を学ぶ。例題がわかりやすいので、ここでの脱落者は皆無。各自で、事実、ルールを拡張する課題を出す。

「リスト処理」

リスト型のデータを説明し、member、append等の述語を試す。はじめは理解しにくい学生もいるが、いろいろな例をためすことによって動きを理解させる。

「パターンマッチング」

単一化の述語 '=' を使う。代入とパターンマッチの違いを理解させる。また、式 $1 + 2$ と値 3 が異なることも確かめさせる。

「再帰プログラム」「ハノイの塔」

フィボナッチ数列などの例を用いて再帰プログラムを説明する。

「カットオペレータと否定」

プログラムの実行を制御するカットオペレータの使い方を学ぶ。

「記号微分」「宣教師と人食土人」

人工知能的な応用プログラムの例。Prologでは簡単に書けるが他の言語ではどうかを考えさせる。

5. まとめ

Prologによる言語処理教育の内容を簡単に説明した。初心者が相手なので難しい理論の話は最少にとどめ、具体的な例題を使うことにより学生の理解を深める方法をとった。

【起動】 A:WORK> PROLOG

【終了】 ?- halt.

【プログラムの読み込み】

?- ['B:SAZAE'].

【エディタの呼出】

?- edit('B:SAZAE,ARI').

【プログラムリスト】

?- listing.

— SAZAE,ARI —

% 事実

男(なみへい).

男(ますお).

男(かつお).

男(たら).

女(ふね).

女(サザエ).

女(わかめ).

父(なみへい, サザエ).

父(なみへい, かつお).

父(なみへい, わかめ).

父(ますお, たら).

母(ふね, サザエ).

母(ふね, かつお).

母(ふね, わかめ).

母(サザエ, たら).

% 規則

親(X, Y) :- 父(X, Y).

親(X, Y) :- 母(X, Y).

祖父(X, Z) :- 父(X, Y), 親(Y, Z).

祖母(X, Z) :- 母(X, Y), 親(Y, Z).

% 以後はコメント

磯野波平は男である

ふぐたサザエは女である

なみへいはサザエの父である

ふねはサザエの母である

父は親である

母も親である

親の父は祖父である

親の母は祖母である

課題

1. サザエさん一家のデータを入力して、様々な質問を試みる。

(例) ?- 母(サザエ, たら).

?- 母(ふね, X).

?- 親(X, サザエ).

?- 祖母(X, Y).

2. 他の親子関係、たとえば、孫・きょうだい・おじ、どのように定義すればよいか考える。

<記号微分のプログラム>

微分を実行するプログラム 微分（式、変数、結果）は次のようになる。

— DIFF.ARI —

微分(X, Y, Z) :- d(X, Y, U), s(U, Z).

d(X, X, 1).

d(C, X, 0) :- atomic(C).

d(U+V, X, DU+DV) :- d(U, X, DU), d(V, X, DV).

d(U-V, X, DU-DV) :- d(U, X, DU), d(V, X, DV).

d(-U, X, -DU) :- d(U, X, DU).

d(K*U, X, K*DU) :- number(K), d(U, X, DU).

d(U*V, X, DU*V+U*dV) :- d(U, X, DU), d(V, X, DV).

s(0*_ , 0).

s(*_0, 0).

s(0+U, A) :- s(U, A).

s(U+0, A) :- s(U, A).

s(1*U, A) :- s(U, A).

s(U*1, A) :- s(U, A).

s(U+V, A+B) :- s(U, A), s(V, B).

s(U-V, A-B) :- s(U, A), s(V, B).

s(U*V, A*B) :- s(U, A), s(V, B).

s(U, U).

$dx/dx = 1$

$dc/dx = 0$

$d(u+v)/dx = du/dx + dv/dx$

...

式の簡略化

このプログラムを使って、次のような例を実行してみる。

?- d(x*x, x, X). ?- 微分(x*x, x, X).

?- d(4*x*x-2*x, x, X). ?- 微分(4*x*x-2*x, x, X).

?- 微分(x*x*x, x, X1), 微分(X1, x, X2), 微分(X2, x, X3).

<問題>

わり算、べき乗、三角関数なども使えるように、このプログラムを拡張しなさい。

それぞれ X/Y X^Y $\sin(X)$ $\cos(X)$ $\tan(X)$ と書くことにする。