

3P-6 初等アセンブラプログラミングを対象とした事例に基づく プログラム評価のための事例ベース構築法

渡辺 博芳

荒井 正之

武井 恵雄

帝京大学理工学部

1. はじめに

我々は、CASLを対象として、提示した課題に対して学生が作成したプログラムの評価を支援するシステムを実現した[1~4]。また、実現したシステムを1999年度の授業で実際に使用した結果、教員の評価作業の軽減に大きな効果があることが明らかになった[3,4]。現在、我々はシステムの拡張と改良を行っている。本稿では、次期システムにおける、事例に基づくプログラム評価処理の効率化と、事例ベース管理処理の容易化を主目的とした事例ベースの構築方法について報告する。特に、評価においてはプログラムリストの一般化は有効であるが、評価事例の事例ベース化においては、CASL文法のまま収納したほうが良いことを指摘し、それらを効果的に管理するための階層的なインデックス構成法を提案する。

2. 事例に基づくプログラム評価

本システムが対象とする評価作業は、学生のプログラムが問題の題意をみたしているかどうかの判定と学生のプログラムに対するアドバイスの作成である。事例に基づくプログラム評価は、「あるプログラムの評価を行う際に、過去の事例を検索し、評価対象のプログラムと同じ実現方法と見なせるプログラムの評価事例が存在すれば、その事例の評価結果を評価対象のプログラムに適用すること」である。このような評価処理のための事例は、(a)プログラムリスト、(b)合否判定結果(合格/不合格)、(c)アドバイス(文)と(d)その他の管理情報から構成する。

3. 事例ベース構築の方針

3.1 従来版の事例ベース構築法の問題点

(1) 事例照合処理の計算コスト大：我々が実現したシステム[1~4]では、事例ベースに事例をフラットに配置し、事例照合処理においては、評価対象のプログ

ラムを事例ベース中の全ての事例と照合する。これでは明らかに効率が悪い。

(2) プログラムリストの一般化表現による問題：1つの事例で複数のバリエーションに対処できるように、事例中のプログラムリストは一般化表現[1]で記述することにした。これによって、事例数削減の効果があるが、一方で以下のような問題も明らかになった。

(a) 本システムのユーザとしての教員が一般化表現を習得する必要がある。

(b) 題意など環境条件の変化[4]が生じると、一般化レベルを調整する必要があり、事例ベース管理処理が複雑になる。

3.2 事例ベース構築方針

前節で述べた問題点を解決するための方針について述べる。まず、前節(1)の問題を改善するために、事例に適切なインデックスを付ける。次に、前節(2)の問題を改善するために、事例のプログラムリストは一般化表現とせずに、CASLの文法に従った表現をとる。事例に含まれるプログラムリストを一般化しない場合、そのままでは1つの事例の適用範囲が狭くなってしまいが、プログラムリストの一般化レベルに基づいて、事例へのインデックスを階層化することで対応する。

プログラムリストの一般化においては、次の3つの問題を解決しておかなければならない。

(1) 命令の種類に関する一般化：一つの操作が2種類以上の命令によって実現可能なことがある。例えば、「あるレジスタに定数を設定する操作」は、CASLでは、LD(load)命令とLEA(load effective address)命令の両方で実現できる。このような使用する命令の違いによるバリエーションをカバーしなければならない。これは、我々が独自に定義した一般化表現と一般化ルール[1]を用いて行うことができる。

(2) 命令の順序に関する一般化：同一の命令群を用いたプログラムでも、個々の命令の記述順序によって実行結果が異なる場合と、変わらない場合がある。この差異をカバーしなければならない。

A Method of Constructing a Case-base for Evaluating
Novice Programs Written in Assembly Language
Hiroyoshi Watanabe, Masayuki Arai and Shigeo Takei
School of Science and Engineering, Teikyo University

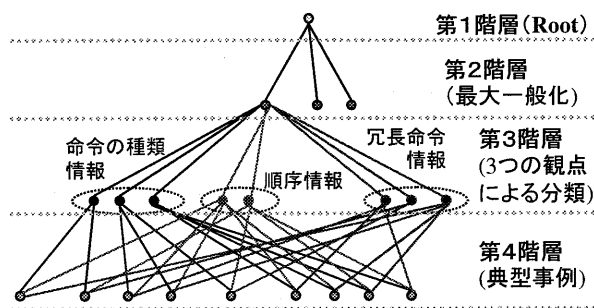


図1: 事例ベースの構造

(3) 冗長命令の有無に関する一般化：ある命令の有無に関わらず、プログラムの動作が同一である場合、その命令は冗長である。冗長命令の有無に関するパリエーションをカバーしなければならない。

4. 事例ベース構築法の詳細

4.1 事例ベースの構造

提案する手法によって構成した事例ベース構造の概要を図1に示す。個々の問題ごとに、図1のようなネットワーク構造の事例ベースを構成する。事例ベースは次の4つの階層から成る。

第1階層 インデックスのルートであり、検索の際のスタート地点となる。

第2階層 プログラムリストを可能な限り一般化したノード(プログラムリストに、本システムが持っている全ての一般化ルールを適用して得られる一般化表現を表すノード)である。

第3階層 3.2で述べた3つの観点ごとに、3種類のノードのグループを形成する。つまり、命令の種類に着目したときに同じと判断される事例は、命令の種類情報グループの同一のノードにリンクする。また、命令の順序関係が同じと判断される事例は、順序情報グループの同一のノードにリンクする。冗長命令についても同様である。このように、1つの事例は個々のグループについて必ず1ノードとリンクされる。

第4階層 典型的な事例を配置する。ここでは、プログラム中のラベル名、レジスタ名が異なる場合も同一プログラムと見なす。

プログラムの同一性を判断する場合に、3.2で述べた3つの観点到優先順を付与することができれば、事例ベースは木構造となるが、事例選択戦略を容易に変更可とするために、上のような構造とした。

第2階層ノード、第3階層の命令の種類情報ノード、第4階層ノード(事例)にはプログラムリストを持たせる。これらのうち、第2階層と第3階層のプログラムリストは一般化表現である。また、それらのプログラ

ムリストに含まれる命令の種類ごとの個数(一般化表現では上限値と下限値)を、各ノードのインデックス情報として持たせる。これによって、照合処理を行う前に、プログラムの照合可能性が判断できる。

第2階層ノードのプログラムリストは最大一般化であり、その下位の事例が必ずしも同じ合否結果になるとは限らない。それに対して、第3階層の命令の種類情報ノードではその下位の事例が同じ合否結果になる程度の一般化を行う。つまり、このノードのプログラムリストは、本システムが持っている一般化ルールのある部分集合を適用して得られる一般化表現である。ここで適用すべき一般化ルールの集合は、評価指示に関するチェックリストにより、教員が指定可能とする。

4.2 事例ベース構築処理の概要

事例ベース構築処理は「ノードの検索」と「ノードの生成および更新」の2つの処理に分けられる。

(1) ノードの検索：新しく登録しようとする事例(新事例)に完全照合する第2階層ノード、第3階層の命令の種類情報ノード、順序情報ノード、冗長命令情報ノード、および第4階層ノードを検索する。検索で得たノードを、それぞれ N_2 , N_{3a} , N_{3b} , N_{3c} , N_4 に代入する。存在しない場合はNULLを設定しておく。

(2) ノードの生成および更新： N_4 が存在する場合、新事例の追加とインデックスの更新は行わない。ただし、新事例と N_4 の情報が競合する場合、新事例の情報を N_4 に反映させる。一方、 N_4 がNULLの場合は、新事例を表す第4階層ノードを生成する。また、 N_2 , N_{3a} , N_{3b} , N_{3c} のうち、NULLであるノードを生成し、関連するノード間にリンクを付加する。

5. おわりに

初等アセンブラプログラミングを対象とした事例に基づくプログラム評価のための事例ベース構築法を提案した。今後、初等アセンブラプログラミング授業支援環境を実現する際に、提案した手法に基づく事例ベース構築機能を組み込む予定である。

参考文献

- [1] 渡辺博芳, 荒井正之, 武井恵雄: CPUとアセンブラ授業のための合否判定支援システム, 情処研報, Vol.97-CE-48, pp.61-68, 1997.
- [2] Watanabe, H., Arai, M. and Takei, S.: Automated Evaluation of Novice Programs Written in Assembly Language, Proc. of ICCE99, Vol.2, pp.165-168, 1999.
- [3] 渡辺博芳, 荒井正之, 武井恵雄: CPUとアセンブラ授業のための事例に基づくプログラム評価支援システム-授業での実用に基づく評価-, 情処学会第60回全大5M-9pp.4-439-440, 2000.
- [4] 渡辺博芳, 荒井正之, 武井恵雄: 事例ベース推論によるプログラム評価支援, 情処研報, Vol.2000-CE-56, pp.9-16, 2000.