

初等アセンブラプログラム評価支援システムの開発と活用

Development and Practical Use of an Evaluation Assistant System of Novice Programs Written in Assembly Language

渡辺博芳 荒井正之 武井恵雄
帝京大学理工学部

〒320-8551 栃木県宇都宮市豊郷台1-1
TEL 028-627-7264 FAX 028-627-7186
E-mail:hiro@ics.teikyo-u.ac.jp

Abstract: A cost-effective educational system is essential for a successful e-learning environment. We developed an evaluation assistant system for novice programs written in an assembly language, CASL. The target evaluation task is to judge whether students' programs satisfy the requirements of the given problem. The system automatically checks the actions of the students' programs using prepared sample data, then the system evaluates the implementation of the programs by using case-based reasoning. This system has been utilized in actual classes for three years, and the results show that the system reduced the teachers' evaluation work drastically.

Keywords: programming education, evaluation assistant systems, COMET /CASL

1. はじめに

コスト・ベネフィットは教育においても重要であり、計算機システムによって自動化できる作業を教員とシステムの協調作業によって効率化し、結果として教育効果を高めることが期待される。このような視点に基づき、我々はCOMET/CASLおよびCOMET /CASL を対象として初等アセンブラプログラム評価支援システム^{[1][2]}を開発し、実際の授業で運用してきた。具体的には、学生が作成したプログラムが出題の題意を満たしているかどうかの判定を支援するものである。本論文では、評価支援システムの全体像を述べるとともに、その効果に関する評価結果について述べる。

2. CPUとアセンブラ演習授業

開発したシステムの支援対象となる授業について説明する。現行のカリキュラムにおける授業時間は、3時間連続の演習授業6回(1.5単位)で、内容は以下の通りである。

- ・最初の授業でCPUとアセンブラプログラミングの概要の講義を受け、シミュレータの使い方を学ぶ。
- ・2回目～5回目は初等アセンブラプログラミング演習であり、簡単な講義の後、課題が提示され、その題意を満たすプログラムを作成する。各課題について合格するまでプログラムの再提出を繰り返す。
- ・6回目は修了試験とCPU理解のための実習である。CPU理解では一つの命令の実行を詳細にトレースするシミュレータの画面を見ながら講義を受け、その後、自分で実際に命令の詳細な実行をトレースして理解を深める。

本演習授業の目的は、一人一人の学生がCPUの構成と動作を正しく理解し、簡単な課題に対して正しく動作す

るプログラムを書けるようになることである。CPUの機能と構成を理解するために、まず、最初に初等アセンブラプログラミングを習得した後、CPUの詳細について学ぶというアプローチをとっている。

この授業が開始された当初から、COMET/CASLを教材として、独自のシミュレータWCASL^[3]を開発し、利用してきた。WCASLは、二つのシミュレーションモードを持つ。初等アセンブラプログラミング習得を目的としたモードでは、1命令の実行でレジスタやメモリがどのように変化するかを見ることが出来る。一方、CPUの動作理解を目的としたシミュレーションモードでは、1命令を実行するフェーズを詳細に見ることが出来る。

3. システムの全体像

(1) プログラムの評価基準

プログラムの評価基準は、「プログラムが正しく動作すること」と「プログラムの実現方法が問題の題意を満たしていること」である。例えば、出題された問題文に「スタックを用いて」という記述が含まれているとき、教員は、学生がスタックについて学ぶことを意図している。この場合、題意通りに動作するプログラムであっても、スタックが使われていなければ、「実現方法は題意を満たしていない」と判断する。

(2) システム構成

現行の初等アセンブラプログラム評価支援システムは、図1のように四つのサブシステムから構成される。

プログラム動作自動評価サブシステム

あらかじめ用意したテストデータを用いて学生のプログラムの動作を自動評価する。プログラムの動作自動評価サブシステムは、Visual C++でWindowsアプリケーションとして開発したもので、学生用クライアントに組み込まれている。

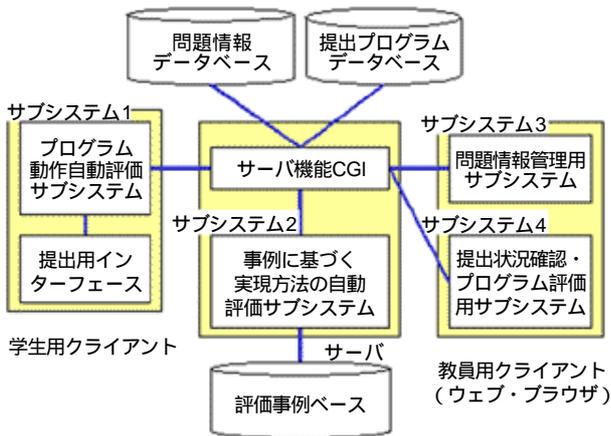


図1 システム構成

事例に基づく実現方法の自動評価サブシステム

評価事例を用いて学生のプログラムの実現方法を評価する。事例に基づく評価とは、過去に評価した評価事例の蓄積の中から、評価しようとするプログラムに類似する評価事例を探し出し、適用可能な場合には、その評価を現在のプログラムの評価とするものである。評価事例は、「評価対象のプログラムリスト」、「評価結果 (accept/reject)」、「アドバイス文」から構成する。学生のプログラムと評価事例との類似度に応じて 評価結果の信頼度を高・低の2段階で出力する。詳細については文献[1]、[2]を参照されたい。

事例に基づく評価サブシステムはコンソールアプリケーションとして開発し、サーバ機能を提供するCGIに組み込んだ。

問題情報管理サブシステム

問題情報は、「問題文」、「ラベルの説明」、「テストデータ」、「キーワード」から構成する。このような問題情報を入力し、管理するサブシステムで、以下のような機能を持つ。

- ア. 問題情報の入力と編集を行う問題情報エディタ機能。
- イ. 既に作成した問題の一覧やキーワードによる検索機能。
- ウ. 授業で出題する問題の提示と提示解除の機能。

提出状況確認・プログラム評価サブシステム

教員が提出状況を確認し、システムの評価結果を参考にしながら、評価を行うために用いるサブシステムである。提出状況確認では、学生のステータスの一覧を参照できる。評価時には、提出順に学生のプログラムが表示される。その際に、そのプログラムに最も類似する過去の評価事例から導かれる評価結果、アドバイス文、それらの信頼度、評価事例内のプログラムリストも併せて表示される。教員は必要に応じて評価結果やアドバイス文を修正することで、最終的な評価を行う。信頼度が高い場合はほとんど修正する必要はない。

教員用のサブシステムである と は、その機能をサーバにCGIとして実現した。教員は、Webブラウザから、これらの機能を利用できる。

(3) 評価処理の流れ

図2に評価処理の流れを示す。学生がプログラムを提出すると、最初に動作の評価を行い、動作が正しくない

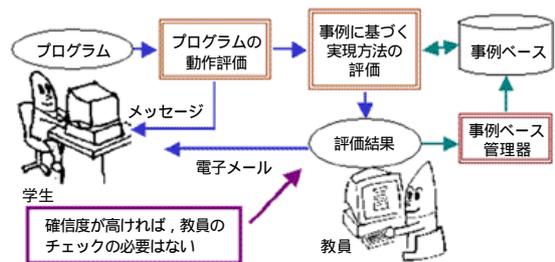


図2 プログラム評価処理の流れ

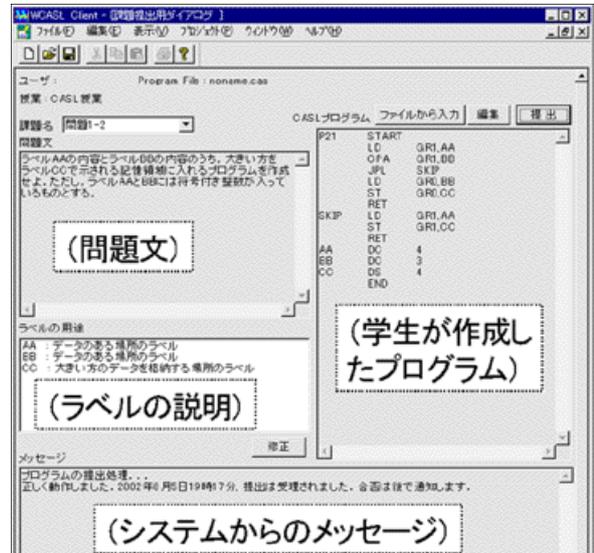


図3 学生用クライアントにおける提出用インターフェース

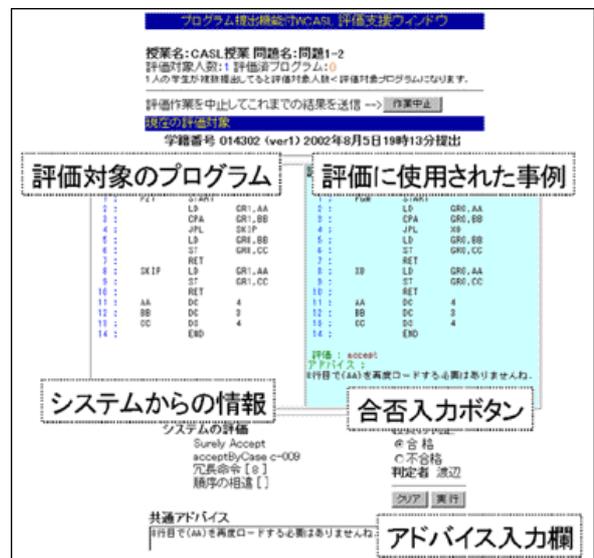


図4 教員がプログラムを評価するためのインターフェース
場合、正しく動作しないデータの例とともに、再提出のメッセージを直ちに提示する。動作が正しい場合は、提出が受け付けられる。

教員が評価を行う際、システムは、学生のプログラムを教員に表示する前に事例に基づく評価を行う。教員はシステムの出力(評価結果)を編集し、最終的な評価を行う。その結果は自動的に電子メールで学生に通知されるとともに、事例ベース更新のための情報として利用する。

図3は学生がプログラムを提出するウィンドウの構成、図4は教員が評価を行うためのインターフェースを示している。

4. 自動評価システム導入の効果

(1) 評価作業効率化の定性的評価

本システムの導入前と導入後の授業を比較する。

システム導入前

授業中は、プログラムができた学生に手をあげてもらい、教員が学生のパソコンのところに行って、評価を行う。学生にプログラムの説明を求め、その間に教員は、そのプログラムが正しく動きそうか、題意を満たしているかについて評価する。評価基準を満たし、学生の説明が的外れでなければ合格とする。

この方式には、「学生に説明をさせるので、学生が自分のプログラムの内容について理解を深めることができる」、「学生の反応を直接に見ながら、アドバイスをすることができる」などの利点がある。

一方、以下のような欠点も存在する。

- ・人間の目でプログラムが正しく動くかどうか判断するので、動作の正しさを適切に評価できない場合もある。
- ・学生とのやりとりにはそれなりの時間がかかるので、全学生の評価に時間がかかる。そのため、学生の評価待ち行列ができてしまう。
- ・教員の労力が評価作業にとられてしまい、つまづいている学生にアドバイスを与える機会が少なくなる。

本システム導入後

学生は専用の提出用インターフェースを使ってプログラムを提出する。正しく動作しない場合には即座にフィードバックが与えられる。教員は評価用インターフェースから評価をする。

本システムの導入で、以下のような利点を享受できる。

- ・教員の評価を行う作業負荷が比較的小さい。そのため、評価作業を一人の教員が行い、他の教員は学生へのアドバイスに専念することができる。
- ・学生の待ち時間は少ない。特に動作が正しくないプログラムの場合は、フィードバックを即座に得られる。また、動作の正しいプログラムを提出すれば、結果を待ちながら、次の課題にとりかかることができる。
- ・評価作業における教員の単純ミスが減少する。

一方、「面談方式でないので、学生に説明をさせたり、反応を直接に見ながらアドバイスができない。」という欠点が存在する。

上記の欠点を補うため、実際には、いくつかの課題については、従来の面談方式で評価を行っている。ただし、この場合でも、プログラムの動作評価は自動的に行えるので、導入前に比較して、教員の作業負荷が小さく、評価作業での単純ミスもほとんどなくなる。

宿題でも、学生は動作の正しくないプログラムについては即座にフィードバックを得ることができる。導入前に比較して学生に再提出を課す回数が減った感がある。

(2) 評価作業効率化の定量的評価

1999年度から2001年度の3年間のシステム利用実績を図5に示す。2001年度からはCASL版である。図中の数字はプログラム数を表す。図5から、提出されたプログラムの約35～50%が動作の評価で自動的に再提出と

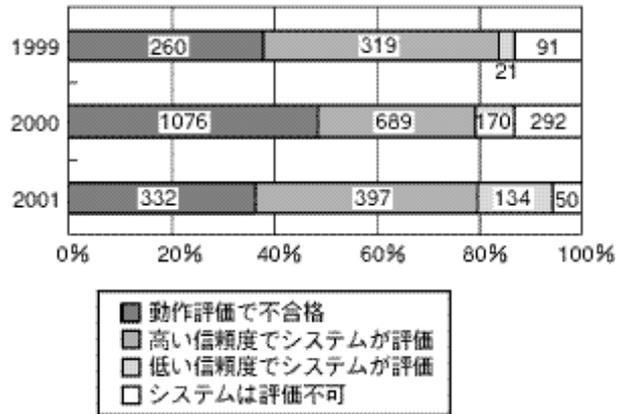


図5 過去3年間の利用実績

表1 実現方法に対する合否判定精度

	1999年度	2000年度	2001年度
全体	99.4%	97.6%	93.4%
高信頼度のみ	100%	99.3%	99.2%

判定されており、さらに約30%～50%が高い信頼度で評価が行われた。現状では、システムが高い信頼度で評価した場合も、念のため教員がチェックをしているが、これら2ケースでは教員が評価を行わずに済むとすれば、約80%の評価作業を軽減できることになる。

(3) システムの判定精度

動作評価の判定精度は、テストデータをきちんと用意できれば、100%となる。この場合の精度は人間である教員が評価する場合よりも高い。一方、事例に基づく実現方法の評価精度を表1に示す。実現方法の評価においても教員はケアレスミスも犯すことを考慮すると、信頼度が高い場合の判定精度は人間である教員と同等といえることができる。

5. おわりに

CASLおよびCASLを対象とした初等アセンブラプログラム評価支援システムについて述べた。過去3年間の利用実績を基に、定性的・定量的に評価を行った結果、本システムが評価作業の効率化に非常に大きい効果があることが明らかになった。

本システムのサーバとクライアントの通信はhttpを用いているので、学生の自宅からプログラムを提出できる。したがって、本システムをセルフラーニング型のインターネット授業で利用することも可能である。

参考文献および関連URL

- [1] 渡辺博芳, 荒井正之, 武井恵雄: 事例に基づく初等アセンブラプログラミング評価支援システム, 情報処理学会論文誌 42 No.1, pp.99-109, 2001.
- [2] Watanabe,H., Arai,M. and Takei,S.: Case-Based Evaluation of Novice Programs. Proc. of AI-ED 2001, pp.55 -64, 2001.
- [3] <http://www.vector.co.jp/soft/win95/prog/se184950.html>

本システムの開発および改良に関する研究の一部は、科学研究費補助金(11680400, 12780293), 人工知能研究振興財団研究助成(12AI320-1)による。