

N-022

プログラミング演習における答案診断のための プロファイル生成システムの実現

Implementation of a Program Profiler for a Computer-Aided Programming Exercise System

岩間 信介†
Shinsuke Iwama

高橋 直久†
Naohisa Takahashi

1 はじめに

我々は繰り返し学習 [1] を支援するプログラミング演習システム CAPES[1][2] の研究を進めている。ここで、本稿が対象としている学習は指導者が提示した問題の題意を満たすプログラムを受講者に作成させるものである。

CAPES では、答案と正解プログラムの実行時の出力結果を比較し、答案の妥当性の評価を即座に行うことで繰り返し学習を支援する。この評価法では次に示す問題点がある。

問題点 1 受講者が実行結果を見てもプログラムの間違い箇所が分からない場合がある

問題点 2 関数や制御構造に間違いがあっても正解と判定されてしまう場合がある

そこで、上記の問題点を解決するためにプロファイル生成システムを提案する。ここで、プロファイルとはプログラムの実行過程から指導者が指定した検査シンボル定義表 (以後、CSD と呼ぶ) を用いて検査シンボルの値の変化と参照関係を構造化したものである。CSD では変数や関数、制御構造などの検査シンボル名とその種類、それらが定義される場所が指導者によって指定される。

2 答案診断のためのプロファイル

答案と正解プログラムの実行時の出力結果を比較する方法では、最終的な出力結果だけを比較するため、出力結果を得るまでの構文や関数の動作が考慮されないこととなる。そのため、構文や関数に誤りがあっても、誤りを発見することができない場合がある。

提案システムにより生成される答案診断のためのプロファイルは、プログラム実行時に指導者が指定した構文や関数内における変数の値が変化していく様子を記録し

たものである。この答案診断のためのプロファイルには、実行結果だけではなく表 1 に示す項目が含まれている。また、プロファイルに含まれる項目により確認できる内容についても表 1 に示す。

表 1: プロファイルに含まれる項目の一覧

含まれる項目	項目より確認できる内容
関数名	関数の参照関係
引数の値	
戻り値	
繰り返し構造の種類	繰り返し構造の中で問題で指定した変数を使って繰り返しを制御していること
変数の名前	
STEP 毎の変数の値	問題で指定された条件分岐を使用していること
条件分岐の種類	
配列名	配列に代入される値の順序と内容
配列内の要素の値	
構造体名	問題で指定されたメンバを用いていること
メンバ名	

表 1 の項目を用いることで、答案プログラムが満たすべき性質を問題で指定することが可能となる。

例として、繰り返し構造 for を制御変数 i で繰り返しを制御するプログラムを作成しなさいという問題を考える。この問題に対する答案プログラムのプロファイルから、答案プログラムが繰り返し構造 for を使用し制御変数 i を用いて繰り返し構造を制御していることが確認できる。また、指導者が想定した正解例プログラムのプロファイルと答案プログラムのプロファイルを比較することにより、答案プログラムの制御変数 i の変化の形が正解例プログラムと同じであることを確認できる。これにより正解例と違う場合には、その違いに応じて指導者が個別に受講者を指導することができる。

†名古屋工業大学大学院工学研究科

3 提案システムの実現法

本章では、提案システムの実現法について述べる。提案システムでは、以下のSTEPによってプログラムよりプロファイルを生成する。

STEP1 プログラム解析機能

ソースプログラムを解析し静的解析データを得る。

STEP2 実行過程生成コマンド作成機能

静的解析データとCSDにより実行過程生成コマンドを作成する。

STEP3 実行過程解析機能

実行過程生成コマンドと入力データ、入力されたプログラム、CSDよりプロファイルを生成する。

各機能の実現法について図1の例を用いながら述べる。

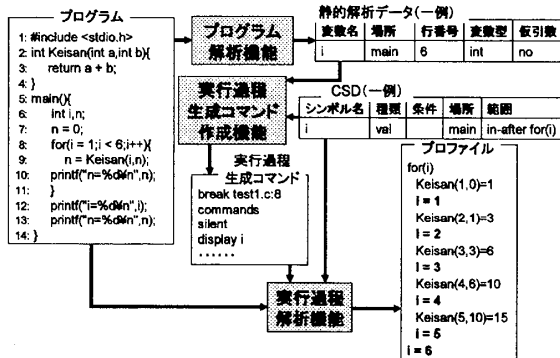


図1: プロファイル生成システムの実行例

3.1 プログラム解析機能

提案システムに入力されたプログラムを解析し、プログラムに記述されている変数や関数、制御構造の名前や種類と記述されている行番号などを静的解析データとして抽出する。

図1では、プログラムから関数 `main` の中で `int` 型の変数 `i` が6行目に定義されていることを図1のような形式で抽出している。

3.2 実行過程生成コマンド作成機能

CSDで指定された検査シンボル名を用いて静的解析データから行番号などを抽出する。そして、抽出したデータを用いて、実行過程解析時に検査シンボルの種類に応じた値の出力を行うことを指定した実行過程生成コマンドを作成する。

図1のCSDでは、関数 `main` にある変数 `i` を制御構造 `for` 以降の値を検査すると指定しているため、このCSDにより変数 `i` の値を出力することを指定している実行過程生成コマンドが作成されている。

3.3 実行過程解析機能

実行過程生成コマンドと入力データを用いてプログラムをデバッガ上で実行して動的解析データを得る。次に、動的解析データよりCSDを用いて該当する検査シンボルの値を抽出する。そして、抽出した値をプログラムの振る舞い分かるように構造化を行いプロファイルを生成する。

図1では、変数 `i` に関する実行過程生成コマンドを用いてプログラムをデバッガ上で動作させ、変数 `i` の値を含んだ動的解析データを得る。そして、動的解析データよりCSDを用いて変数 `i` の値を抽出しプロファイルを生成している。

4 プロトタイプシステム

提案システムより生成されるプロファイルの有効性を検証するためにプロトタイプシステムを実装した。プロトタイプシステムでは、表1で示したプロファイルが含む項目の内、関数名から配列内の要素の値までの項目について実装を行った。また、実行過程解析機能において動的解析データを得るデバッガとしてGDB[3]を使用した。

配列データをソートさせる問題を例題として取り上げた。この問題で、ソートした結果は正しいと判定された答案プログラムに対してプロトタイプシステムを適用させ、プロファイルを生成した。

プロファイルを確認すると、答案プログラムにおける配列の要素の値の変化のさせ方が分かった。これにより、答案プログラムの幾つかは問題で指定されたアルゴリズムとは違った方法でソートを行っていることが分かった。以上のことより、プロファイルにより答案プログラムの問題点の発見が容易であることが分かった。

5 おわりに

本稿では、プログラミング演習における答案診断のためのプロファイル生成システムを提案しプロトタイプシステムを実装した。今後はプロファイルを用いて演習問題に対する正誤判定や誤り箇所に関する情報の提示について検討を進める。

参考文献

- [1] 中島秀樹, 高橋直久, 細川宜秀: プログラミング演習のためのQAサイクル-受講者の習得度に応じた問題提示メカニズム-, 電子情報通信学会論文誌, VOL.J88-D-I, NO.2, pp439-450(2005).
- [2] 中島秀樹, 宮地恵佑, 高橋直久: プログラミング演習支援システムCAPESのための答案評価機構の実現, 情報処理学会 研究報告, 2006-CE-83(18), pp127-134(2006).
- [3] Richard M.Stallman, Roland H.Pesch (コスモ・プラネット訳): GDB デバッギング入門, アスキー, 東京, 1992.