

プログラミング演習における模範解答を用いた テストケース評価基準の自動生成

尾崎 裕樹 榊原 一紀 中村 正樹 富山県立大学

1 はじめに

プログラミング演習において、学生自身がテストケースを設計しテストを行うことによって、学生が作成したプログラムを自身で評価することができる。文献 [1] では、テストケースが適切であるかを判定する評価基準を用意して、設計したテストケースの評価基準に対する網羅率を求めることで、テストケースを評価する手法が提案されている。このシステムでは、教員が演習問題ごとにテストケース評価基準を与える必要がある。

本研究では、テストケース評価基準作成時における教員の負担を減らすために、テストケース評価基準を自動生成する手法を提案する (図 1)。

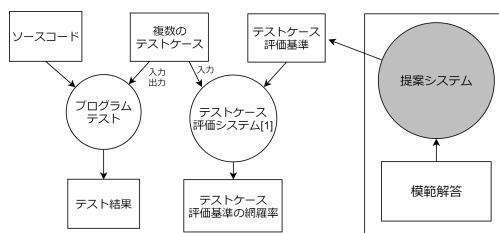


図 1: 提案システムの位置づけ

2 テストケース評価基準の自動生成

プログラミング演習では、通常のソフトウェア開発と異なり教員が模範解答を用意しているという点に着目し、提案システムでは、テストケース評価基準を生成する際に、問題文 (仕様) を分析して記述するのではなく、模範解答 (ソースプログラム) を解析し、入力変数と条件式を抽出することによって、テストケース評価基準を生成する。2つの整数値を受け取り、大きさを比較した結果を表示するプログラムを例にテストケース評価基準を生成する手順を説明する。まず、変数宣言部で変数の型と名前を構造体に格納し、標準入力命令部で使用される変数の型に基づき基準を生成する。例えば以下の模範解答の int 型の入力変数に対し、

```

変数宣言
int num1; int num2;
標準入力命令
scanf("%d", &num1); scanf("%d", &num2);

```

提案システムは次のような評価基準を生成する。

```

入力のデータ構造
(int num1 int num2)

```

```

入力変数から生成されたテストケース評価基準
num1 > 0, num1 == 0, num1 < 0
num2 > 0, num2 == 0, num2 < 0

```

次に、条件文を抽出することでテストケース評価基準を生成する。その際、分岐網羅をするために、条件文の否定も評価基準に含める。以下の模範解答の条件文に対し、

```

条件文の一部
if (num1 < num2){
    printf("%dより%dの方が大きい。\\n",
        num1, num2);
} else if (num1 > num2){
    printf("%dより%dの方が大きい。\\n",
        num2, num1);
} else{
    printf("2つの数は同じ値。\\n");}

```

提案システムは次のような評価基準を生成する。

```

条件文から生成されたテストケース評価基準
num1 < num2, !(num1 < num2)
num1 > num2, !(num1 > num2)

```

3 プログラミング課題への適用

参考書 [2] の 5 章 (場合に応じた処理) までのサンプルコードや練習問題のうち、標準入力があるプログラムを対象に、作成したシステムによってテストケース評価基準の生成が可能か検証した。検証を行ったプログラムは 26 個であり、そのうちの 17 個で生成可能であった。生成できなかったプログラムは getchar で入力を行っているプログラムと、同じ変数に複数回入力しているプログラム、switch 文や条件演算子で場合分けをしているプログラムであった。

また、8 章で扱う関数を含むプログラムは入力変数の名前と関数の引数の名前が同じ場合のみ評価基準を生成することができた。

4 おわりに

本研究では、学生が自身でテストケースの評価を行うために用いるテストケース評価基準を自動生成する方法を提案した。現在対応できていない配列や繰り返し文、関数を含むプログラムからの評価基準の自動生成が今後の課題である。

参考文献

- [1] 蜂巢ら: プログラミング演習におけるテストケース評価システム, コンピュータソフトウェア, vol.34, no.4, pp.54-60, 2017.
- [2] 高橋麻奈: やさしい C, SB クリエイティブ, 2012.