

ソフトウェアテスト学習のためのプログラミング 演習システムの自動テスト機能の拡張

杉本 優太*1・北 英彦*1・高瀬 治彦*1

Email: 419m225@m.mie-u.ac.jp

*1: 三重大学大学院工学研究科

◎Key Words プログラミング演習, ソフトウェアテスト, テストケース

1. はじめに

近年, 社会の情報化が著しく進む中, プログラミング教育の重要性が増している。しかし, 実践的なソフトウェア開発において重要なソフトウェアテストに関する知識を学ぶ機会は少ない。そこで, 学者向けにソフトウェアテストについて学習できるシステムの作成を試みる。

本研究では, プログラミングの学習の際にソフトウェアテストに関する概念をプログラミングと同時に学習できる環境を構築することをめざす。一般的なソフトウェア開発において, 作成したソースコードが正常に動作するかどうかを確認するソフトウェアテストは大切であるが, 前述のようにプログラミング教育の際にはあまり触れられない。そこで, 本研究では, ソフトウェアテストの中でも重要な概念であるテストケースの作り方・テストの網羅率について学習するための機能・コンテンツを研究室で開発・運用中のプログラミング演習システム PROPEL^①に組み込むことで, 演習中にソフトウェアテストについて学べるようにする。

2. ソフトウェアテスト技法

学習対象であるソフトウェアテストに関して紹介する^②。ソフトウェアテストは技法によって主にブラックボックステストとホワイトボックステストの2つに分類される。本章ではソフトウェアテストの必要性と2種類の技法について紹介する。

2.1 ソフトウェアテストの必要性

ソフトウェア開発においてソフトウェアテストの重要性は高い。システム開発における開発からリリースまでの流れに関して, 開発工程とテスト工程の対応関係を表すモデルの1つに図1に示すV字モデルがある。

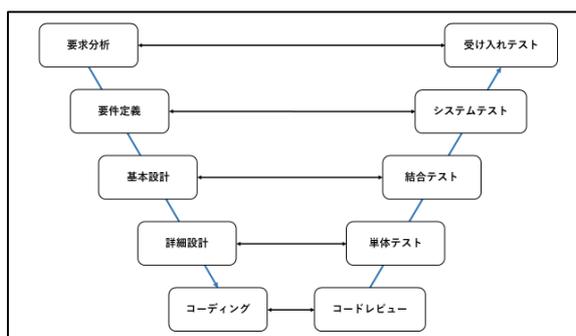


図1 システム開発でのV字モデル

品質の高いソフトウェアを開発するためには, 図1に示すように開発工程の各工程に対応するテストを行う必要がある。プログラミング学習で行うのは詳細設計・コーディングの部分なので, 単体テストについて学習する必要がある。

2.2 ソフトウェアテスト技法の分類

テストを実行する際にはテストケースが必要となる。テストケースとは, 「入力」と「(その入力に対して期待される) 出力」の組み合わせのことである。ソースコードの動作を確認するために, 複数のテストケースを作成してテストを行う。テストケース作成技法の観点で, ブラックボックステストとホワイトボックステストの2種類に分類される。

2.2.1 ブラックボックステスト

ブラックボックステストはソースコードの中身を見ずに行うテストであり, 入力に対して期待される出力がされるかどうかを確認するテストである^③。システムの仕様を満たしているかを確認するテストであるため, 「ユーザー側のテスト」と言われている。ブラックボックステストには主に, 入出力の関係の集合をいくつかの同値クラスに分類し, その代表値をテストケースとする同値分割法と, 同値分割法で得られた同値クラスの端の値とその前後の値をテストケースとする境界値分析法の2種類の方法がある。本研究では, 学習者には手始めとして境界値分析法を学習させる。

2.2.2 ホワイトボックステスト

ホワイトボックステストはソースコードの中身を見ながら, テストケースを作成するテストである^④。ソースコードの構造やロジック, 制御の流れの成否を確認するため, 「作り手側のテスト」と言われている。ホワイトボックステストには, 命令網羅テスト (C0), 分岐網羅テスト (C1), 条件網羅テスト (C2) の順に厳しい網羅基準がある。本研究では, 初学者を対象としているため基本的かつ最低限達成すべき網羅基準であるC0・C1を学習対象とする。実際のソフトウェア開発においても, この二つの網羅率が100%であることが求められている。

3. ソフトウェアテスト学習システムの提案

2章で述べたソフトウェアテストの必要性を考慮し、初学者が講師の指導なしにソフトウェアテストについて学ぶことができる自己学習システムの開発を提案する。また、本システムは三重大学工学部電気電子工学科2年生の授業にて運用されている、初学者向けプログラミング演習システムに組み込むものとする。

3.1 自己学習システムの要件

一般に、授業におけるプログラミング学習では、V字モデルの「詳細設計→コーディング」の部分の演習が行われる。しかし、単体テストは学習者に任されている。そのため、講師が演習後に学習者が作成したソースコードをチェックしたときに、十分なテストを行っていないものが多々見受けられる。本研究では、学習者が適切な単体テストを実行できるように、分岐網羅テスト（ホワイトボックステスト）および境界値テスト（ブラックボックステスト）を学習できるようにする。

自己学習システムの要件を以下のように定める。

- ・ ソフトウェアテストの必要性を学習できる
- ・ 初学者でも理解しやすい
- ・ 学習者のみで学習できる
- ・ 分岐網羅テストの実践に取り組める
- ・ 境界値テストの実践に取り組める

3.2 自己学習の流れ

学習者は以下の流れで学習に取り組むこととする。

1. Web教材による自己学習
2. 実践課題（ブラックボックステスト）
3. 実践課題（ホワイトボックステスト）
4. 教材学習後の理解度確認テスト

3.2.1 Web教材による知識の自己学習

初めに、ブラックボックステスト・ホワイトボックステストそれぞれについて解説されたWeb教材を用いて初学者に学習させる。

学習内容は以下にまとめる。

- ・ ソフトウェアテストの目的・必要性
- ・ ブラックボックステスト
- ・ ホワイトボックステスト
- ・ テストケース

3.2.2 実践課題（ブラックボックステスト）

2.2.1で述べたように、本研究では境界値テストを学習対象とする。例えば、課題として「0以上100以下の整数を読み込み、60点以上を合格、60点未満を不合格とせよ。」を例にあげる。テストケースの対を（入力、期待される出力）のように表記することとすると、この課題におけるテストケースの一例としては、(-1, エラー) / (0, 不合格) / (59, 不合格) / (60, 合格) / (100, 合格) / (101, エラー) の6つがあげられる。このように課題を提示し、テストケースを作成させる。

3.2.3 実践課題（ホワイトボックステスト）

2.2.2で述べたように、今回は分岐網羅テスト（C1）を用いる。課題は3.2.2と同じものについて紹介する。この課題におけるソースコードの一例は図3のようになる。

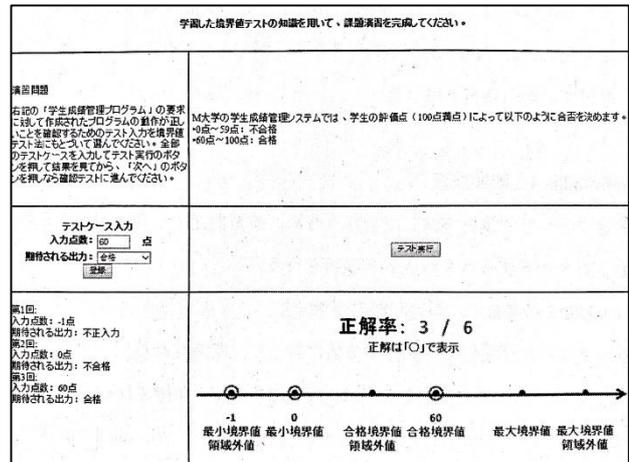


図2 実践課題画面（ブラックボックステスト）

```

2 import java.util.Scanner;
3
4 public class Main {
5     public static void main(String[] args) {
6         Scanner input = new Scanner(System.in);
7         System.out.print("点数を入力してください(0以上100以下の整数):");
8         int score = input.nextInt();
9         input.nextLine();
10        if (score < 0 || score > 100) {
11            System.out.println("点数として正しくありません");
12        } else if (score >= 60) {
13            System.out.println("合格");
14        } else {
15            System.out.println("不合格");
16        }
17    }
18 }
    
```

図3 合格表示のソースコード例

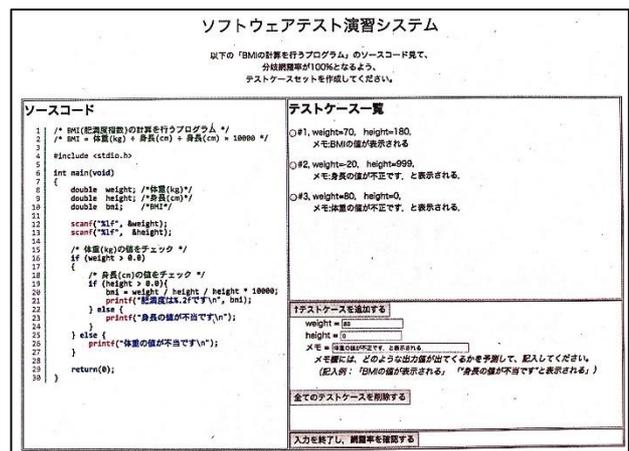


図4 実践課題画面（ホワイトボックステスト）

3.2.4 教材学習後の理解度確認テスト

最後に総復習として、境界値テストおよび分岐網羅テストにおけるテストケース作成の演習を行わせる。ここで、学習者本人がソフトウェアテストに関する知識を習得できているか確認する。

4. まとめ

ソフトウェア開発において重要な単体テストについて、プログラミング初学者が以下の知識を学習者自ら習得できるシステムを提案した。

- ・ ソフトウェアテストの概要
- ・ ブラック/ホワイトボックステストの概要
- ・ 境界値テストにおけるテストケース作成方法
- ・ 分岐網羅テストにおけるテストケース作成方法

本システムは現在未実装であるが完成次第運用し、フィードバックを行うことで著者は本システムのさらなる改善を考えている。

参考文献

- (1) 伊富昌幸, 北英彦, 高瀬治彦, 林照峯: コーディング状況に応じたアドバイスを可能にするプログラミング演習システムに関する研究, コンピュータ利用教育協議会, 2010PC カンファレンス (2010)
- (2) Cem Kaner, Jack Falk, Hung Quoc Nguyen, テスト技術者交流会訳: 基本から学ぶソフトウェアテスト (2001)
- (3) 袁智韜: プログラミング能力向上を目的としたソフトウェアテストの自己学習システムに関する研究, 三重大学大学院工学研究科修士論文, 2016
- (4) 高桑稔: プログラミング能力向上を目的としたプログラムテストの学習システムに関する研究, 三重大学大学院工学研究科修士論文, 2015